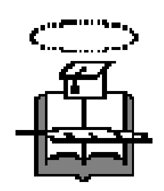


UNIVERSIDAD DE GUAYAQUIL

Facultad de Ciencias Matemáticas y Físicas

Carrera de Ingeniería en Sistemas Computacionales

"SISTEMA DE AUDITORÍA DE SEGURIDADES DE ROUTER Y SWITCH CISCO VIA WEB"

PROYECTO DE GRADO

CURSO DE GRADUACIÓN

Previo a la Obtención del Título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

Autores:

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

GUAYAQUIL-ECUADOR

Año: 2010

AGRADECIMIENTO

Damos gracias a Dios por habernos permitido alcanzar la meta profesional que nos propusimos.

A nuestros padres que con amor y sacrificio acompañaron cada paso de nuestras vidas estudiantiles y nos supieron conducir por el camino de los grandes ideales.

A nuestros amigos más cercanos que nos dieron todo su apoyo de manera incondicional.

A los profesores y compañeros que han iluminado y compartido cada uno de los rincones de nuestras etapas de estudios.

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

DEDICATORIA

Damos gracias a Dios por habernos permitido alcanzar la meta profesional que nos propusimos.

A nuestros padres que con amor y sacrificio acompañaron cada paso de nuestras vidas estudiantiles y nos supieron conducir por el camino de los grandes ideales.

A nuestros amigos mas cercanos que nos dieron todo su apoyo de manera incondicional.

A los profesores y compañeros que han iluminado y compartido cada uno de los rincones de nuestras etapas de estudios.

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

TRIBUNAL DE GRADUACIÓN

Presidente

1er. Vocal

2do. Vocal

Secretario

DECLARACIÓN EXPRESA

"La autoría de la tesis de grado corresponde exclusivamente a los suscritos, perteneciendo a la Universidad de Guayaquil los derechos que generen la aplicación de la misma"

(Reglamento de Graduación de la Carrera de Ingeniería en sistemas Computacionales, Art. 26)

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

RESUMEN

A continuación detallaremos lo que es una auditoria de router y/o switch, este puede ser de software o hardware y es aquel que comprueba la información procedente de Internet o una red y a continuación, deniega o permite el paso de ésta al equipo, en función de la configuración del dispositivo. De este modo, me ayuda a impedir que los hackers y software malintencionado obtengan acceso al mismo.

La seguridad ha sido el tema principal a tratar cuando una organización desea conectar su red privada al Internet. Sin tomar en cuenta el tipo de negocios, se ha incrementado el número de usuarios de redes privadas por la demanda del acceso a los servicios de Internet, ya que es un medio que nos permite estar comunicados tanto externamente como internamente.

Se desarrollara un sistema que haga posible la comparación de datos extraídos de un router y/o switch contra las buenas políticas de seguridad, que los administradores de red han establecido de acuerdo a las necesidades de la organización.

Se ha determinado las buenas políticas mediante entrevistas a algunos expertos del área. Debido a que los administradores de red tienen que desarrollar todo lo

concerniente a la seguridad de sus sistemas, ya que se expone la organización privada de sus datos así como la infraestructura de su red a los Expertos de Internet.

El sistema a desarrollar tendrá la capacidad de analizar y almacenar los datos extraídos en una base de datos, manejar perfiles de administrador y auditor con sus debidos permisos o restricciones.

Generara reportes y los resultados de las auditorias de router y/o switch.

Lo más importante del sistema a desarrollar es que podrá dar sugerencias al auditor acerca de las vulnerabilidades del router emitiendo el respectivo reporte y recomendación para mitigar dichas vulnerabilidades. Cabe recalcar que la decisión la toma los administradores de las areas de sistemas de las empresas auditadas.

INDICE

Agradecimiento II

Dedicatoria III

Tribunal de Graduación IV

Declaración Expresa V

Resumen VI

Índice VIII

Índice de Figura XIII

1.- ENTORNO PARA EL DESARROLLO DEL SIATEMA

1.1 Antecedentes……………………………………………………………...- 7–

1.2 Problemática…………………………………………………………- 7-

1.3 Solución a Problemática……………………………………………...- 8-

1.4 Misión………………………………………………………….......- 8-

1.5 Visión………………………………………………………….......- 9-

1.6 Objetivos…………..………………………………………………- 9-

1.6.1 Objetivos Generales…………………………………………… ...- 9-

1.6.2 Objetivos Específicos.……………………………………........- 10-

1.7 Alcances……………………………………………………………....- 12-

1.8 Beneficios de la Herramienta……………………………………....- 13-

1.9 Metodología de Desarrollo…………………………………………...- 14-

1.9.1 Análisis……………………………………………………….....- 15-

1.9.2 Diseño…………………………………………………………………..- 15-

1.9.3 Programación………………………………………………………………- 15-

1.9.4 Pruebas………………………………………………………………- 16-

1.9.5 Implementación………………………………………………………- 17-

1.10 Herramientas para Desarrollar la Aplicación……………………………….- 17-

1.10.1 Hardware…………………………………………………………- 17-

1.10.2 Software……………………………………………………………….- 18-

1.10.3 Humano……………………………………………………………..- 18-

1.11 Requerimiento para que funcione la Aplicación……………………………..- 18-

1.11.1 Hardware…………………………………………………………- 18-

1.11.2 Software……………………………………………………………….- 18-

1.12 Cronograma……………………………………………………………….- 19-

2. ANALISIS DEL SISTEMA………………………………………………..- 20-

2.1 Propósito del Análisis………………………………………………………- 20-

2.2 Levantamiento de Información……………………………………………..- 20-

2.3 Análisis de Requerimientos…………………………………………………- 24-

2.4 Análisis de Herramientas de Desarrollo…………………………………….- 24-

2.5 Representación de la Arquitectura…………………………………………- 27-

2.5.1 Diagrama de Entidad-Relación………………………………………- 27-

2.5.2 Especificación de Objetos……………………………………………- 38-

2.5.3 Diagrama de Flujos de Datos…………………………………………- 42-

2.5.4 Especificaciones de Procesos…………………………………………- 45-

2.5.5 Diagrama de Transición de Estados…………………………………- 52-

2.5.6 Diccionario de Datos………………………………………………- 59-

3. DISEÑO DEL SISTEMA……………………………………………..- 71-

3.1 Propósito del Diseño……………………………………………………...- 71-

3.2 Menú Principal……………………………………………………....- 71-

3.2.1 Procesos………………………………………………………- 72-

3.2.1.1 Conectar a Router y/o Switch……………………………...…- 72-

3.2.1.2 Auditoria de Router y/o Switch……………………………….- 72-

3.2.1.3 Exportar Datos………………………………………….…- 72-

3.2.2 Consultas…………………………………………………….....- 73-

3.2.2.1 Extracciones Realizadas…………………………………....- 73-

3.2.2.2 Auditorias Anteriores…………………………………………- 73-

3.2.3 Reportes………………………………………………………- 73-

3.2.4 Mantenimiento……………………………………………….- 74-

3.2.4.1 Compañía……………………………………………………..- 74-

3.2.4.2 Usuarios……………………………………………………-74-

3.2.4.3 Cliente……………………………………………………...- 74-

3.2.4.4 Router y/o Switch…………………………………………….- 75-

3.2.4.5 Secuencia…………………………………………………...- 75-

3.2.4.6 Cambio de Clave……………………………………………...- 75-

3.2.4.7 Actualiza Clave……………………………………………….- 76-

3.2.5 Acerca de………………………………………………………- 76-

3.3 Diseño de Interfaz Grafica de Usuario……………………………………..- 76-

3.3.1 Conexión de Base de Datos…………………………………………….- 76-

3.3.2 Menú Principal…………………………………………………………- 77-

3.3.3 Crear Empresa…………………………………………………………..- 79-

3.3.4 Crear Dispositivo……………………………………………..............- 80-

3.3.5 Crear Marcas………………………………………………………..…..- 81-

3.3.6 Crear Router y/o Switch………………………………………….…- 82-

3.3.7 Crear Auditoria……………………………………………………......- 83-

3.3.8 Crea y Añade Dispositivo……………………………………………...- 84-

3.3.9 Crear Comandos……………………………………………………….- 85-

3.3.10 Crear Protocolos y Puertos…………………………………………- 86-

3.3.11 Crear Usuarios……………………………………………………….. - 87-

3.3.12 Crear Roles……………………………………………………………. ..- 88-

4. DESARROLLO DEL SOFTWARE……………………………………. -89-

4.1 Codificación de lo Principales Componentes………………………………..-89-

4.1.1 Proceso de Conexión a Router o Switch……………………………… .- 89-

4.1.2 Proceso de Obtención de Información………………………………... - 90-

4.1.3 Proceso de Verificación………………………………………………..- 90-

4..1.4 Proceso de Asignación……………………………………………....- 90-

4.2 Desarrollo de Pruebas e Implementación…………………………….… ..- 91-

4.2.1 Creación de la Base de Datos……………………………………… …- 91-

4.2.2 Pruebas del Sistema…………………………………………….. …- 91-

4.2.2.1 Pruebas de Aplicación Ensambladas……………………………….- 91-

4.2.2.2 Pruebas de Aplicación con Varios Usuarios……………………….- 91-

4.1 Implementación del Sistema……………………………………………….- 91-

4.1.1 Componentes del Software…………………………………………… - 91-

4.1.2 Componentes del Hardware………………………………………….. - 92-

5. CONCLUSIONES Y RECOMENDACIONES……………….…… ….-93-

4.1 Recomendaciones………………………………………………………….. - 93-

4.1 Conclusiones……………………………………………………………… … - 94-

Anexos………………………………………………………………………… -95-

Bibliografía………………………………………………………………….. -99-

INDICE DE FIGURA

Figura 1 Cronograma de trabajo…..-19-

Figura 2 Estructura de Entrevista……………………………………….-21-

Figura 3 DER……………………………………………………..……..-37-

Figura 4 Especificación de Objetos……………………………………..-41-

Figura 5 DFD Nivel 0……………………………………………..……..-42-

Figura 6 DFD Nivel 1..-43-

Figura 7 DFD Nivel 2…………………………………………….…..-44-

Figura 8 Presentación del Sistema……………………………………….-77-

Figura 9 Menú Principal……………………………………….……...-78-

Figura 10 Crear Empresas ……………………………………………….-79-

Figura 11 Crear Dispositivos ……………………………………….……-80-

Figura 12 Crear Marcas…………………………………………………...-81-

Figura 13 Crear Modelos de Router y/o Switch……………………………-82-

Figura 14 Crear Auditoria…………………………………………………..-83-

Figura 15 Crea y Añade Dispositivo………………………………………..-84-

Figura 16 Crear Comandos…………………………………………….……-85-

Figura 17 Crea Protocolo y Puertos…………………………………………-86-

Figura 18 Crear Usuarios …………………………………………….……..-87-

Figura 19 Crear Roles……………………………………………………......-88-

Figura 20 Base de Datos PostgreSQL 8.4 ……………………………………-.89-

Figura 21 Datos extraídos del Router o Switch Cisco…………………….….-90-

INTRODUCCION

Resaltamos la importancia de los Routers y Switchs CISCO, así como, que una buena configuración de estos equipos ayuda a mitigar los riesgos de acceso de intrusos, robo o alteración de la información, manteniendo de esta forma la integridad de los datos y por ende de la empresa debemos de tener en cuenta la importancia de la auditoría de estos, procurando examinar los temas pertinentes a la revisión de los dispositivos antes y después de ser asegurados.

Esta herramienta permite al auditor de Sistemas, automatizar sus procesos de auditoría de la configuración de los Routers y los Switchs beneficiándolo, para de esta forma llevar un mejor control de la administración de estos equipos.

El router es analizado con más detenimiento, teniendo en cuenta la importancia de las posibilidades que provee, mientras que el switch es estudiado desde un punto de vista más físico que lógico.

Es importante resaltar que el tratamiento del aseguramiento es un tema muy importante en la actividad de un administrador de seguridad, ya que permite

identificar las vulnerabilidades de los dispositivos y por ende desarrollar las herramientas y medidas necesarias para minimizar los riesgos ante posibles amenazas.

Permitirá diagnosticar los valores de seguridad de un equipo Router, con base en la aplicación de reglas basadas en la configuración de seguridades de estos equipos. Ayudando al auditor a emitir de forma más fácil recomendaciones de mejoras de seguridad. Así como también facilita la obtención de consultas a partir de la información que nos proporciona el Router y el Switch; permitiendo analizar las actividades de los usuarios y de diagnostico de estos.

La metodología usada para la elaboración de este proyecto se basa en el modelo de cascada con retroalimentación el mismo que se fundamenta en el análisis, diseño, implementación y pruebas. Por medio de este modelo es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático.

El sistema a desarrollar tendrá la capacidad de analizar y almacenar los datos extraídos en una base de datos, implementar seguridades de acceso al sistema,

manejar perfiles de administrador y auditor con sus debidos permisos o restricciones. Generara reportes del análisis, y los resultados de la auditoria.

Creara una bitácora de los diversos accesos al mismo para que sean utilizados como pistas de Auditoria.

Lo más importante del sistema a desarrollar es que podrá dar sugerencias al auditor acerca de las vulnerabilidades de los Routers y los Switchs emitiendo los respectivos reportes y recomendaciones para mitigar dichas vulnerabilidades. Cabe recalcar que la decisión la toma el auditor de sistemas.

En la actualidad las empresas necesitan conectividad tanto internamente como externamente, es por este motivo que se debe tener implementado las mejores prácticas en los equipos que permiten dicha conectividad.

Es complicado para el administrador de red verificar constantemente si los Routers poseen la mejor configuración debido a la gran cantidad de tiempo que se toma la verificación de cada uno de ellos.

Las compañías de auditoría están siempre tratando de obtener una ventaja competitiva. Esto ocasiona que la tecnología sea una parte fundamental para que se puedan alcanzar metas de forma eficiente y eficaz. Estas compañías auditoras se han dado cuenta de la necesidad de automatizar sus procesos para disminuir costo, tiempo y es aquí donde se debe tener a la tecnología como aliada, ya que por medio de esta podemos automatizar procesos que en la actualidad se hacen manualmente y de esta forma poder ofrecer auditorias continuas.

A continuación detallaremos lo que es una auditoria de router y/o switch, este puede ser de software o hardware y es aquel que comprueba la información procedente de Internet o una red y a continuación, deniega o permite el paso de ésta al equipo, en función de la configuración del dispositivo.

De este modo, me ayuda a impedir que los hackers y software malintencionado obtengan acceso al mismo.

La seguridad ha sido el tema principal a tratar cuando una organización desea conectar su red privada al Internet. Sin tomar en cuenta el tipo de negocios, se ha incrementado el número de usuarios de redes privadas por la demanda del acceso a

los servicios de Internet, ya que es un medio que nos permite estar comunicados tanto externamente como internamente.

Se desarrollara un sistema que haga posible la comparación de datos extraídos de un router y/o switch contra las buenas políticas de seguridad, que los administradores de red han establecido de acuerdo a las necesidades de la organización.

Se ha determinado las buenas políticas mediante entrevistas a algunos expertos del área. Debido a que los administradores de red tienen que desarrollar todo lo concerniente a la seguridad de sus sistemas, ya que se expone la organización privada de sus datos así como la infraestructura de su red a los Expertos de Internet.

El sistema a desarrollar tendrá la capacidad de analizar y almacenar los datos extraídos en una base de datos, manejar perfiles de administrador y auditor con sus debidos permisos o restricciones.

Generara reportes y los resultados de las auditorias de router y/o switch.

Lo más importante del sistema a desarrollar es que podrá dar sugerencias al auditor acerca de las vulnerabilidades del router emitiendo el respectivo reporte y recomendación para mitigar dichas vulnerabilidades. Cabe recalcar que la decisión la toma los administradores de las áreas de sistemas de las empresas auditadas.

CAPITULO # 1

1. ENTORNO PARA EL DESARROLLO DEL SISTEMA

1.1. Antecedente.

Las compañías de auditoría están siempre tratando de obtener una ventaja competitiva. Esto ocasiona que la tecnología sea una parte fundamental para que se puedan alcanzar metas de forma eficiente y eficaz. Estas compañías auditoras se han dado cuenta de la necesidad de automatizar sus procesos para disminuir costo, tiempo y es aquí donde se debe tener a la tecnología como aliada, ya que por medio de esta podemos automatizar procesos que en la actualidad se hacen manualmente y de esta forma poder ofrecer auditorias continuas.

1.2. Problemática

En la actualidad las empresas necesitan conectividad tanto internamente como externamente, es por este motivo que se debe tener implementado las mejores prácticas en los equipos que permiten dicha conectividad.

Los problemas que podríamos encontrar son de acuerdo a la política seguridades, normas de acceso a la red, hay o no conexión remota, reglas de encriptación, protección de virus, la configuración entre hardware de nuestros Routers y Switchs Cisco. Qué cantidad de puertos primordiales están activados o desactivados, Cuantas veces ha sido utilizado el Router y el Switch.

1.3. Solución A Problemática

La Solución del caso sería la creación de un aplicativo vía web que permita dar la facilidad tanto al administrador como al auditor teniendo en cuenta las necesidades que tienen cada uno de los departamentos dependiendo de sus configuraciones y siguiendo un estándar.

1.4. Misión.

Crear una herramienta efectiva para el auditor de Redes que permita verificar datos de los Routers y Switchs Cisco que dependiendo de la misma pueda arrojar como resultado un informe de recomendaciones de las mejores prácticas aplicables a dichos equipos.

1.5. Visión.

Llegar a ser líderes en el mercado de herramientas de auditoría brindado una alternativa tecnológica eficiente y eficaz, capaz de entregar información clave sobre Routers y Switchs Cisco en un portal web.

1.6. Objetivos.

La aplicación permite al auditor realizar un diagnostico de los parámetros de seguridad de forma automatizada, y mostrar las recomendaciones necesarias para cumplir con las mejores prácticas de seguridad en una red.

1.6.1. Objetivos Generales.

- Desarrollar una herramienta capaz de diagnosticar valores de seguridad en Routers y la cobertura de éste con el switch.

- Identificar si los parámetros de seguridad cumple con las normas de auditoria

- Restablecer los canales de comunicaciones en sus diferentes puntos.

- Mantener la supervisión , ajuste remotos y locales de los switch a través del Router

- Reducir los riesgos de comunicación entre los diferentes puertos

- Ofrecer una interfaz de Usuario amigable que sea fácil de usar parar el auditor y que sirva para automatizar los procesos de auditoría de los Routers y Switchs.

1.6.2. Objetivos Específicos.

- Auditar la configuración de los puertos del Switch en este caso nuestro proyecto ésta orientada a un Switch Catalyst 24160 de 24 puertos de 10/100 con 2 puertos de 1Gb.

- Auditar la configuración del Router 2801 serie 2800 Cisco

- El sistema manejara políticas de seguridades para los usuarios (Administrador y Auditor)

- Extraer la información Router y Switch (CISCO) hacia el sistema de auditoría que usaremos mediante el SysLog de estos.

- Identificar si los parámetros de seguridades cumplen con las buenas políticas (estándares) de auditoria

- Comparar la información obtenida vs. las buenas políticas (estándares) de auditoria

- Obtener consultas a partir de la información obtenida del Router y del Switch (CISCO).

- Utilizar un Dash Board para ver la información de forma dinámica en línea al momento de la extracción

- Genera reportes de operaciones realizadas a uno o varios Usuarios.

- Utilizar una Base de Datos para el manejo de la información.

1.7. Alcances.

- La herramienta tendrá un proceso de inicio de sesión. Este proceso de seguridad permitirá el acceso o denegación a la aplicación.

- Manejara dos tipos de perfiles de usuario Administrador y Auditor

- Opción para la creación de clientes.

- Permitir la interconexión Router y Switch de la familia CISCO para extraer la información de los parámetros de configuración de los clientes.

- Evaluación automática de los estándares de configuración obtenidos del (Router y Switch) contra la estándares de configuración previamente guardados en nuestra Base de Datos.

- Muestra las vulnerabilidades del Router y Switch las respectivas recomendaciones para mitigar algún percance futuro.

- Genera reportes planos y gráficos del análisis y los resultados de la auditoria de Router y Switch (CISCO).

- Bitácora del uso, manejo y accesos al sistema.

1.8. Beneficios de la Herramienta.

- Optimizar el trabajo del Auditor en Sistemas de Información, disminuyendo el tiempo dedicado a esta labor.
- Reportes que ayudaran a identificar las vulnerabilidades del Router y/o Switch
- Emisión de reportes detallados para la alta gerencia
- Brinda seguridad al auditor sobre los reportes que el realiza, ya que cada usuario tiene su código de acceso especifico.
- Ahorra tiempo en la extracción y comparación con las buenas prácticas.
- Facilita la realización de auditorías continuas

1.9. Metodología de desarrollo.

1.9.1. Análisis.

En esta etapa analizamos los requerimientos de la aplicación y las necesidades que este requiere para funcionar. Para esta etapa realizamos las siguientes actividades:

- Levantamiento de Información.- Obtenemos la información necesaria por medio de entrevistas con administradores de centro de cómputo, investigación del funcionamiento de los Router, Switch y de su configuración.
- Identificación de necesidades.- Se obtiene por medio del análisis del planteamiento del proyecto, de las entrevistas e investigaciones.

- Análisis de herramientas necesarias.- Elección de Base de Datos, Elección de lenguaje de programación. En base a las opciones e interfaz del aplicativo.

- Análisis de la estructura de la aplicación.- Elaboración del Diagrama Entidad Relación, Diagrama de transición de estados, Diagrama de flujo de datos, Especificación de procesos, Diccionario de datos.

1.9.2. Diseño.

Se diseña la estructura necesaria para el funcionamiento del aplicativo.

Tenemos las siguientes actividades:

- Diseño de opciones del menú
- Diseño de Interfaz de usuario de la aplicación.
- Diseño de reportes de la aplicación.

1.9.3. Programación

En esta etapa se procede a desarrollar la aplicación, en base al análisis este software va a ser desarrollado en Java, Itext, y con la base de datos PostGreSql. Se realizaran dos actividades relevantes:

- Creación de la estructura de la aplicación.- Realizar la interfaz grafica y la estructura de la Base de Datos.
- Desarrollo de la aplicación web.

1.9.4. Pruebas

Durante la prueba del sistema, el sistema se emplea de manera experimental para asegurarse de que el software no tenga fallas, es decir, que funciona de acuerdo con las especificaciones y en la forma en que los usuarios esperan que lo haga.

Se alimentan como entradas conjunto de datos de prueba para su procesamiento y después se examinan los resultados.

Las pruebas a realizar se especifican a continuación:

- Verificaron y Validación
- Pruebas de seguridad

- Control de Calidad

- Pruebas de Unidad

- Ejecución de la aplicación en tiempo real.

1.9.5. Implementación.

Para realizar la implementación de la aplicación seguiremos los siguientes pasos:

- Realizar con Hibernate la creación de Base de Datos
- Correr el Ingreso de parámetros a la Base de Datos
- Realiza arquitectura físicamente de la Red

1.10. Herramientas Para Desarrollar La Aplicación

1.10.1. Hardware

3 PC´s 80 Gb, 2Gb RAM, Dual Core 1.6 Ghz.

1.10.2. Software

Base de Datos PostgreSql

Lenguaje de Programación Java

1.10.3. Humano

3 desarrolladores con sueldo de $ 450 mensuales por 5 meses: $6750

1.11. Requerimientos Para Que Funcione La Aplicación

11.1. Hardware

2 PC´s 80 Gb, 2Gb RAM, Dual Core 1.6 Ghz

1.11.2. Software

Base de Datos PostGreSql

Eclipse Galileo y Ganimede para Java

1.12. Cronograma.

Nombre de tarea	Duración	Comienzo	Fin
Sist. Aud. Router Switch Cisco	**101 días**	**mar 16/06/09**	**sáb 31/10/09**
Analisis	**37 días**	**mar 16/06/09**	**mar 04/08/09**
Analisis de planteamiento del Proyecto	10 días	mié 17/06/09	lun 29/06/09
Investigacion a cerca de protocolos de comunicación	3 días?	mar 30/06/09	jue 02/07/09
Investigacion a cerca de servicios del Router	2 días	vie 03/07/09	lun 06/07/09
Investigacion acerca de la configuracion de un Router Cisco	3 días	mar 07/07/09	jue 09/07/09
Eleccion del Router Fisico	16 horas	vie 10/07/09	lun 13/07/09
Investigacion sobre la configuracion del Switch	5 días	vie 03/07/09	jue 09/07/09
Eleccion del Switch Fisico	16 horas	vie 10/07/09	lun 13/07/09
Entrevista con administradores de redes	2 días	mar 14/07/09	mié 15/07/09
Eleccion de B.D. a utilizar	6 horas	vie 17/07/09	vie 17/07/09
Configurar los estandares de seguridad	3 días	vie 17/07/09	mar 21/07/09
Escogitamiento del Lenguaje de Programacion	6 horas	mié 22/07/09	mié 22/07/09
Establecer Opciones para el Sistema	4 días	mié 22/07/09	lun 27/07/09
Establecer los Estandares Web para el sistema	3 días	mié 29/07/09	vie 31/07/09
Analizar la Estructura de nuestra B.D.	2 días	lun 03/08/09	mar 04/08/09
Diseño	**8 días**	**mié 05/08/09**	**vie 14/08/09**
Diseño del G.U.I. del Sistema Web	4 días	mié 05/08/09	lun 10/08/09
Diseño de Reportes que se aplicaran	4 días	mar 11/08/09	vie 14/08/09
Programaciòn	**41 días**	**lun 17/08/09**	**lun 12/10/09**
Desarrollo la Estructura de la B.D.	6 días?	lun 17/08/09	lun 24/08/09
Desarrollo del G.U.I. Del Sistem Web	5 días?	mar 25/08/09	lun 31/08/09
Desarrollo de Seguridades de Acceso al Sistema Web	5 días?	mar 01/09/09	lun 07/09/09
Desarrollo de Perfiles en gestion de Usuarios	5 días?	mar 08/09/09	lun 14/09/09
Desarrollo de Conexiones con el Router y el Switch	5,5 días?	mar 15/09/09	mar 22/09/09
Desarrollo en Verificar la obtencion de Datos	7 días?	mar 22/09/09	jue 01/10/09
Desarrollo de Codigo en Java	5 días?	jue 01/10/09	jue 08/10/09
Creacion de Reportes Finales	3 días?	jue 08/10/09	mar 13/10/09
Pruebas	**10 días**	**mar 13/10/09**	**mar 27/10/09**
Verificacion	3 días?	mar 13/10/09	vie 16/10/09

Figura 1 Cronograma de Trabajo

CAPITULO # 2

2. ANÁLISIS DEL SISTEMA.

2.1. Propósito del Análisis

El propósito del Análisis es el de ayudar a los auditores de sistemas de información, para que estos puedan sacar una buena auditoria quiere decir todo lo relacionado con la configuración, estándares y funcionamientos del equipo Router junto con su Switch en este caso nuestro proyecto está situado en algunos elementos de la familia cisco especificados en la introducción.

2.2. Levantamiento de Información.

Para el levantamiento de información hemos consultado por la internet,libro y también haciéndole preguntas a profesionales especializados en CISCO (entrevistas) específicamente en la parte de la configuración y su funcionamiento

Las entrevistas que realizamos se basan en preguntas abiertas y cerradas es una gran ayuda para nuestro proyecto ya que con esto podemos establecer un mecanismo a seguir para realizarlo con estas necesidades

La estructura que se utilizara en la entrevista es:

Embudo.

Comienza la entrevista con preguntas abiertas y termina con preguntas cerradas.



Figura 2 Estructura entrevista

Entrevista para Establecer las Buenas Prácticas de la Auditoria

1. ¿Cuál es su nombre completo?

2. ¿Cuánta Experiencia tiene Administrando Routers y Switch CISCO?

3. ¿Qué cursos a realizado para administración de Routers y SWITCH CISCO?

4. ¿Cuántos equipos de hardware de comunicación hay en la empresa?

5. ¿De qué marca y modelo son sus equipos?

6. ¿Alguien le recomendó los equipos que utiliza su empresa, si es así

¿Nos podría usted decir bajo que parámetros lo hizo?

7. ¿Qué tipo de software utilizan sus equipos?

8. ¿Qué tipos de políticas de seguridad utiliza su empresa para la extracción de datos específicamente de un equipo router?¿ Cuales son los objetivos claves de estas políticas?

9. ¿Cuáles son los parámetros para establecer las políticas de seguridad? Y Porque? ¿Cuáles son los riesgos que se intentan mitigar con estas políticas?

10. ¿Existe alguna relación entre la topología de red con la manera que usted maneja la seguridad de su empresa?

11. ¿Cuáles son los procedimientos que usted realiza para evaluar la seguridad de sus equipos de comunicación. ?

12. En caso de problemas, ¿Cuales han sido las medidas o pasos tomados para mitigar los riesgos con los equipos de comunicación?

13.¿ Estadísticamente si es posible, podría determinar el comportamiento de la seguridad de sus equipos de comunicación? Siempre seguros, % de problemas por año, tipos de problemas, recurrencia de problemas por tipo, etc.

14. ¿De qué forma usted está segura de que las políticas de seguridad de los equipos son actualizadas convenientemente y usted las conoce?

15. ¿Me podría explicar detalladamente cada parámetro de configuración de seguridad del equipo, sus riesgos inherentes y las ventajas de hacerlo?

16. ¿Podría usted describir otras instalaciones que usted conozca donde se apliquen este tipo de procedimientos o se utilicen equipos similares?

17. ¿Le gustaría implementar un software de auditoría de Router en su empresa?

Se espera mediante estas entrevistas discernir las buenas prácticas de configuración de un switch físico. **Ver Anexo**

2.3. Análisis de Requerimientos

El análisis de requerimientos es la tarea que plantea la asignación de software a nivel de sistema y el diseño de programas.

En la herramienta de Auditoria de Routers se han detectado tres requerimientos que se detallan a continuación:

Proceso de Seguridad.- Este proceso permitirá el acceso o denegación a la aplicación. Utiliza dos perfiles Revisor y Consultor.

Proceso de Administración.- Este proceso permite la configuración de la herramienta así como la inserción de nuevas políticas de configuración de Router y Switch.

Proceso de Consultas y Reportes.- Este proceso permitirá realizar los reportes de auditoría del Router, y emitirá recomendaciones de mejoras de seguridad.

2.4. Análisis de Herramientas de desarrollo

En esta parte del documento se describe porque se ha utilizado las siguientes herramientas de desarrollo.

Base de Datos PostGreSql .- Nos ayuda construir aplicaciones robustas y fiables ofreciendo una sencilla pero potente base de datos que es además gratuita y de fácil comunicación con java.

Ventajas

- Ideal para pequeñas instalaciones de servidor y aplicaciones de escritorio con requerimientos más elevados, como búsquedas de texto completo o procesado de consultas XML.

- La descarga, desarrollo, instalación y redistribución son gratuitas

- Las bases de datos puede ser variable en capacidad.

Herramienta de Desarrollo de Sistemas JAVA.-

Este lenguaje es muy efectivo ya que todo me lo permite llevar en objetos haciéndolo más eficiente y eficaz utilizaremos la versión de java 6.0 con su formulario GALILEO

Ventajas

- Posee varias bibliotecas para manejo de base de datos, pudiendo conectarse con cualquier base de datos por medio de Persistencia.

- Permite un desarrollo eficaz y menor inversión en tiempo que con otros lenguajes.

Herramienta para crear reportes Itext.-

Es la solución de elaboración de informes más usada en el mundo.
Una arquitectura común para acceso a datos, generación de informes y distribución de información, que permite responder con rapidez a cualquier necesidad de generación de informes o desarrollo de aplicaciones. Ha sido diseñado para integrarse de forma sólida con los recursos de aplicaciones, web y datos ya existentes, sin imponer estándares y procesos.

Ventajas

• La carga de informes con datos guardados es mucho más rápida, pudiendo a empezar a visualizar el informe antes de finalizar la carga total del mismo.

• Posibilidad de creación de informes en tiempo de diseño como en tiempo de ejecución, permitiéndole al usuario final una máxima personalización de los mismos.

2.5. Representación de la Arquitectura.

Para la representación de la arquitectura de nuestra herramienta usaremos algunos componentes del análisis de la Programación Estructurada.

Diagrama Entidad – Relación.

Especificaciones de Objetos

Diagrama de Flujos de Datos

Especificación de Procesos

Diagrama de Transición de Estados

Diccionario de Datos

2.5.1. Diagrama Entidad – Relación

En base al relevamiento realizado mediante las entrevistas realizadas, en investigaciones se ha determinado, la necesidad de manejar repositorios de datos para las siguientes entidades.

auditorias.- En esta tabla se almacenan los datos de cuando se crea una auditoria

auditorias
ID_AUDITORIA
ID_EMPRESA
FECHA_CREACION
FECHA_AUDITORIA
OBSERVACION_CREACION
OBSERVACION_AUDITORIA
ID_USUARIO

comandos.- En esta tabla se almacenan la descripción de los comandos con sus recomendaciones

comandos
ID_COMANDO
ID_TIPO_COMANDO
COMANDO
MENSAJE1
MENSAJE2
HABILITADO

comandos_modelos .- En esta tabla intermedia se almacena los datos comandos por modelos.

comandos_modelos
ID_COMANDO
ID_MODELO

detalles_auditorias.- Almacena los detalles de la extracción

detalles_auditorias
ID_DETALLE_AUDITORIA
ID_AUDITORIA
ID_DISPOSITIVO_EMPRESA
CONFIGURACION

detalles_auditorias_comandos.- Almacena la comparación de la extracción con los comandos establecidos para la extracción

detalles_auditorias_comandos
ID_DETALLE_AUDITORIA_COMANDO
ID_DETALLE_AUDITORIA
ID_COMANDO
CUMPLIO_COMANDO

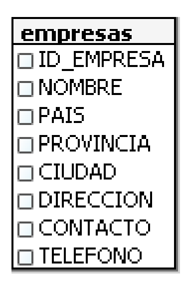
detalles_listas_acceso.- Almacena los datos de las listas de acceso que van hacer comparadas.

detalles_listas_acceso
ID_DETALLE_LISTA_ACCESO
ID_LISTA_ACCESO
HABILITADO
TIPO_PROTOCOLO
IP_ORIGEN
WILD_CARD_ORIGEN
IP_DESTINO
WILD_CARD_DESTINO
OPERADOR
PUERTO
NOMBRE_SERVICIO

dispositivos_empresas- Almacena los datos de los dispositivos de cada empresa con sus respectivos passwords para la extracción de los mismos.

dispositivos_empresas
ID_DIPOSITIVO_EMPRESA
ID_EMPRESA
ID_MODELO
IDENTIFICADOR
DEPARTAMENTO
COMENTARIO
IP
PASSWORD_TELNET
PASSWORD_MODO_PRIVILEGIADO
USUARIO

empresas.- Almacena los datos con la identificación completa de las empresas.



interfaces_red.- Almacena los datos de cada interfaz de los dispositivos que van hacer auditados .

interfaces_red
ID_INTERFAZ
ID_DETALLE_AUDITORIA
NOMBRE_INTERFAZ
DIRECCION_IP
MASCARA_SUBRED
IP_RED

interfaces_red_politicas_trafico.- Almacena los datos de la buenas prácticas que se utilizaran para auditorias posteriores.

interfaces_red_politicas_trafico
ID_INTERFAZ_RED_POLITICA_TRAFICO
ID_INTERFAZ
ID_POLITICA_TRAFICO
CUMPLIO

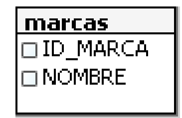
listas_acceso.- Almacena los datos de las listas de acceso para las auditorias del sistema.

listas_accesos
ID_LISTA_ACCESO
ID_DETALLE_AUDITORIA
NOMBRE_LISTA_ACCESO

listas_acceso_interfaces_red.- Almacena los parámetros para ver en qué sentido va la lista de acceso.

listas_accesos_interfaces_red
ID_LISTA_ACESO_INTERFAZ_RED
ID_INTERFAZ
ID_LISTA_ACCESO
SENTIDO

marcas.- Almacena la marca del dispositivo.



modelos.- Almacena los modelos de los dispositivos sea switch o router.

modelos
ID_MODELO
ID_MARCA
NOMBRE
ID_TIPO_DISPOSITIVO

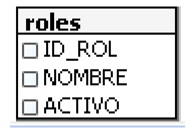
políticas_trafico.- Almacena los datos de la creación de los puerto que van hacer auditados con sus recomendaciones.



políticas_trafico_modelo.- tabla intermedia para cada puerto para cada dispositivos.

politicas_trafico_modelos
ID_POLITICA_TRAFICO
ID_MODELO

Roles.- Almacena los datos la identificación de cada usuario que manejan el sistema.



tipos_comandos.- Almacena los comandos que va hacer comparados en el servidor.



tipos_dispositivos.- En la tabla almacena el nombre con el tipo de dispositivos.

tipos_dispositivos
ID_TIPO_DISPOSITIVO
NOMBRE

tipos_politicas_traficos.- Almacena los tipos de políticas que utilizan los dispositivos en el servidor.

tipos_politicas_traficos
ID_TIPO_POLITICA_TRAFICO
NOMBRE

usuarios.- Almacena los datos del usuario del servidor.

usuarios
ID_USUARIO
USUARIO
PASSWORD
NOMBRES
APELLIDO_PATERNO
APELLIDO_MATERNO
ACTIVO

usuarios_roles.- Almacena los usuarios y quien le designo el rol usuario sea auditor o administrador.

usuarios_roles
ID_USUARIO_ROL
ID_USUARIO
ID_ROL
ID_USUARIO_ASIGNO_ROL
FECHA_INICIO_VIGENCIA
ID_USUARIO_DESASIGNO_ROL
FECHA_FIN_VIGENCIA

De las entidades descritas anteriormente se determino el siguiente modelo entidad – relación.

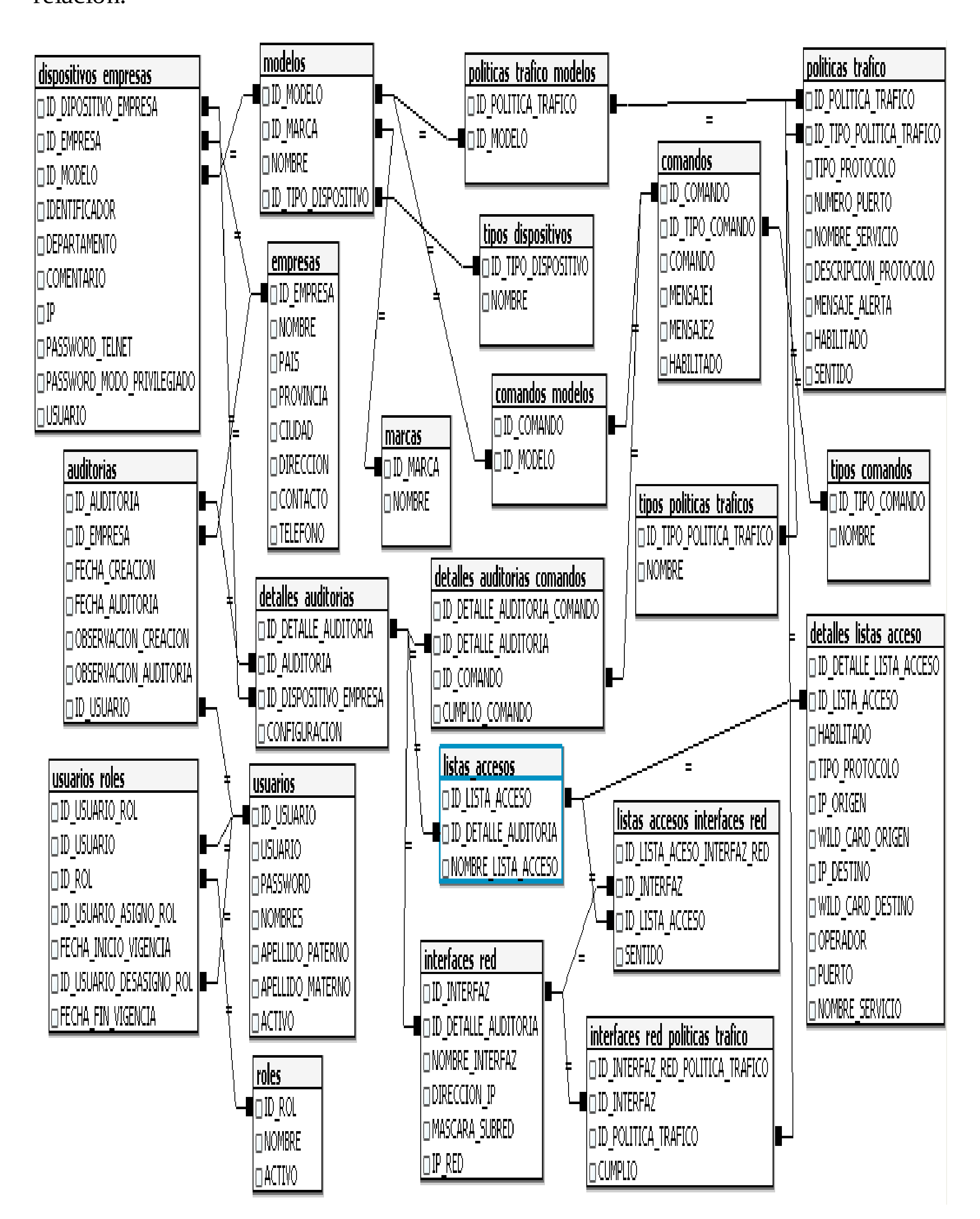


Figura 3 DER

2.5.2. Especificaciones de objetos

auditorias
ID_AUDITORIA
ID_EMPRESA
FECHA_CREACION
FECHA_AUDITORIA
OBSERVACION_CREACION
OBSERVACION_AUDITORIA
ID_USUARIO

comandos
ID_COMANDO
ID_TIPO_COMANDO
COMANDO
MENSAJE1
MENSAJE2
HABILITADO

comandos_modelos
ID_COMANDO
ID_MODELO

detalles_auditorias
ID_DETALLE_AUDITORIA
ID_AUDITORIA
ID_DISPOSITIVO_EMPRESA
CONFIGURACION

detalles_auditorias_comandos
ID_DETALLE_AUDITORIA_COMANDO
ID_DETALLE_AUDITORIA
ID_COMANDO
CUMPLIO_COMANDO

detalles_listas_acceso
ID_DETALLE_LISTA_ACCESO
ID_LISTA_ACCESO
HABILITADO
TIPO_PROTOCOLO
IP_ORIGEN
WILD_CARD_ORIGEN
IP_DESTINO
WILD_CARD_DESTINO
OPERADOR
PUERTO
NOMBRE_SERVICIO

dispositivos_empresas
ID_DIPOSITIVO_EMPRESA
ID_EMPRESA
ID_MODELO
IDENTIFICADOR
DEPARTAMENTO
COMENTARIO
IP
PASSWORD_TELNET
PASSWORD_MODO_PRIVILEGIADO
USUARIO

empresas
ID_EMPRESA
NOMBRE
PAIS
PROVINCIA
CIUDAD
DIRECCION
CONTACTO
TELEFONO

interfaces_red
ID_INTERFAZ
ID_DETALLE_AUDITORIA
NOMBRE_INTERFAZ
DIRECCION_IP
MASCARA_SUBRED
IP_RED

interfaces_red_politicas_trafico
ID_INTERFAZ_RED_POLITICA_TRAFICO
ID_INTERFAZ
ID_POLITICA_TRAFICO
CUMPLIO

listas_accesos
ID_LISTA_ACCESO
ID_DETALLE_AUDITORIA
NOMBRE_LISTA_ACCESO

listas_accesos_interfaces_red
ID_LISTA_ACESO_INTERFAZ_RED
ID_INTERFAZ
ID_LISTA_ACCESO
SENTIDO

marcas
ID_MARCA
NOMBRE

modelos
ID_MODELO
ID_MARCA
NOMBRE
ID_TIPO_DISPOSITIVO

politicas trafico
ID_POLITICA_TRAFICO
ID_TIPO_POLITICA_TRAFICO
TIPO_PROTOCOLO
NUMERO_PUERTO
NOMBRE_SERVICIO
DESCRIPCION_PROTOCOLO
MENSAJE_ALERTA
HABILITADO
SENTIDO

politicas trafico modelos
ID_POLITICA_TRAFICO
ID_MODELO

roles
ID_ROL
NOMBRE
ACTIVO

tipos comandos
ID_TIPO_COMANDO
NOMBRE

tipos dispositivos
ID_TIPO_DISPOSITIVO
NOMBRE

tipos politicas traficos
ID_TIPO_POLITICA_TRAFICO
NOMBRE

usuarios
ID_USUARIO
USUARIO
PASSWORD
NOMBRES
APELLIDO_PATERNO
APELLIDO_MATERNO
ACTIVO

usuarios_roles
ID_USUARIO_ROL
ID_USUARIO
ID_ROL
ID_USUARIO_ASIGNO_ROL
FECHA_INICIO_VIGENCIA
ID_USUARIO_DESASIGNO_ROL
FECHA_FIN_VIGENCIA

Figura 4 Especificación de Objetos

2.5.3. Diagrama de flujo de Datos

Diagrama de Nivel 0

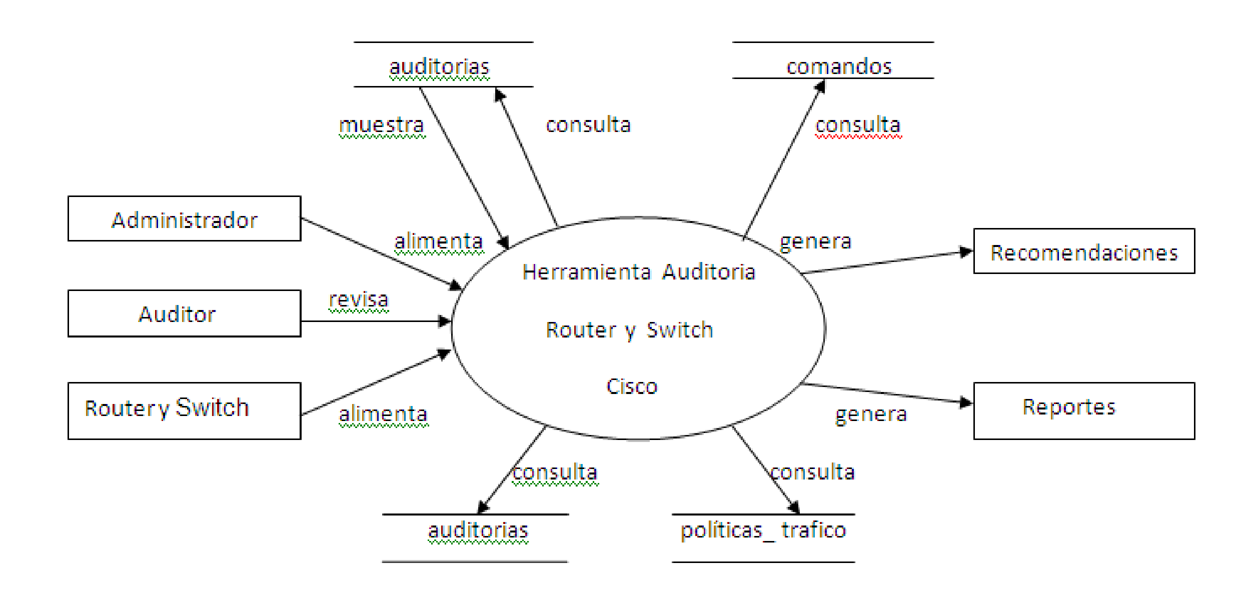


Figura 5 DFD Nivel 0

Diagrama de Nivel 1

Figura 6 DFD Nivel 1

Diagrama de Nivel 2



Figura 7 DFD Nivel 2

2.5.4 Especificación de Procesos.-

Proceso

Validación de Usuario

- ✓ Verifica que usuario existe
- ✓ Verifica contraseña sea correcta
- ✓ Si usuario existe y contraseña correcta Permite acceso al

Proceso

Verificación Privilegios

- ✓ Verificar si el usuario es administrador o auditor
- ✓ Habilita opciones según privilegios

Proceso

Cambio de clave

- ✓ Verifica la clave actual
- ✓ Verifica que la nueva clave cumpla con las políticas de seguridad establecidas por el administrador

Proceso

Conexión a Router y Switch

- ✓ Verifica usuario ,claves, IP del router y switch
- ✓ Valida que exista el router y switch

Proceso

Extracción de Datos

- ✓ Verificar los privilegios del usuario
- ✓ Extraer los datos del router y/o switch por medios de comandos almacenados en el sistema
- ✓ Almacenamiento de los registros extraídos en la base de datos

Proceso

Acceso Sistema

- ✓ Verifica los privilegios del usuario
- ✓ Muestra las diferentes opciones del sistema

Proceso

Selección de Router y Switch

- ✓ Permite escoger el modelo de router y/o switch de los clientes que se van a auditar

Proceso

Selección de Políticas

- ✓ Permite escoger las políticas de seguridad del Router y Switch que se van a evaluar

Proceso

Configuración Router y Switch

- ✓ Comparar datos extraídos vs. Los comandos y políticas trafico

Proceso

Vulnerabilidades

- ✓ Muestra las falencias en la configuración del equipo

Proceso

Recomendaciones

- ✓ Verifica las políticas y resalta las que no cumple con los comandos y políticas trafico

Proceso

Consulta de Auditoria Anteriores

- ✓ Verifica si existe una auditoría realizada anteriormente
- ✓ Muestra los datos de la auditoria anterior

Proceso

Auditoria por Empresa

- ✓ Valida que el cliente exista
- ✓ Genera un reporte con los datos del cliente

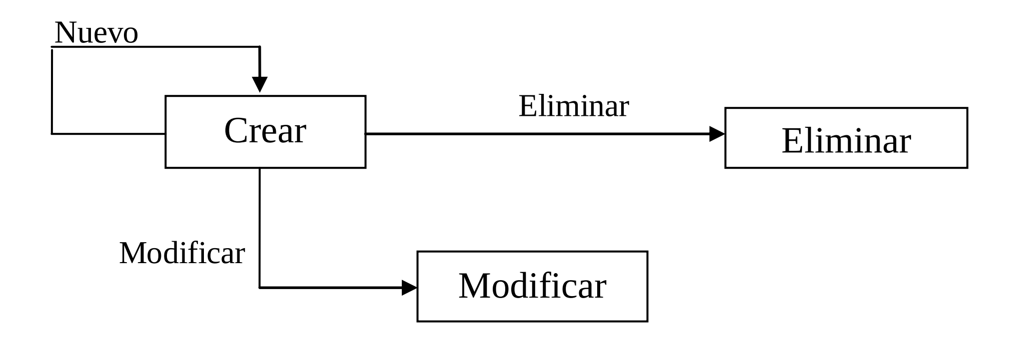
Proceso

Reporte

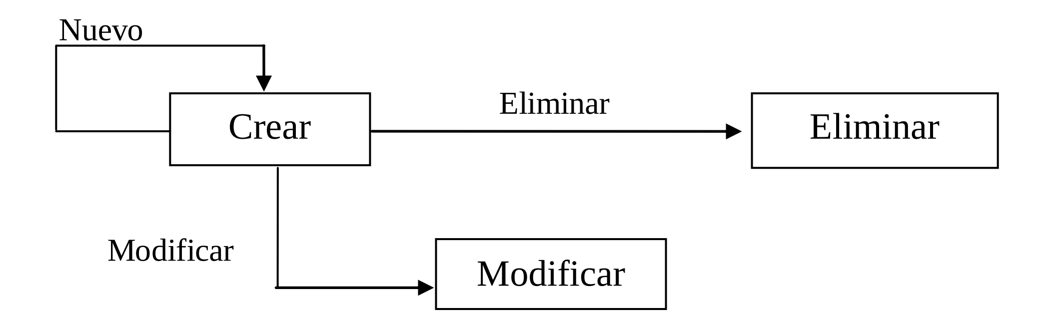
- ✓ Genera reportes en pdf detallado cada vulnerabilidad encontrado
- ✓ Emite una recomendación por cada vulnerabilidad
- ✓ Emite reporte grafico del cumplimentó

2.5.5 Diagramas de Transición de Estados

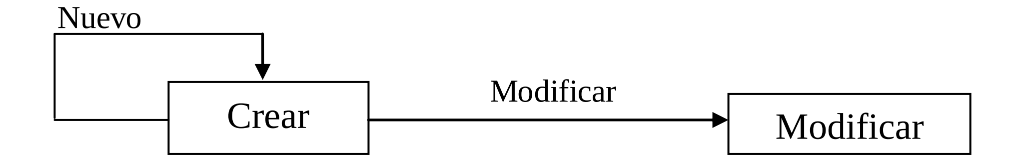
auditorias



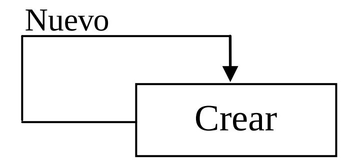
comandos



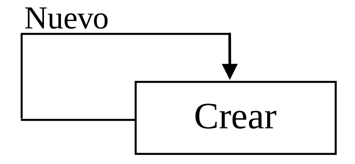
comandos_modelos



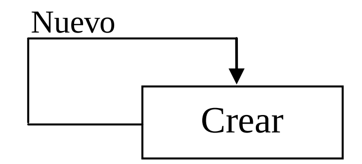
detalles_auditorias



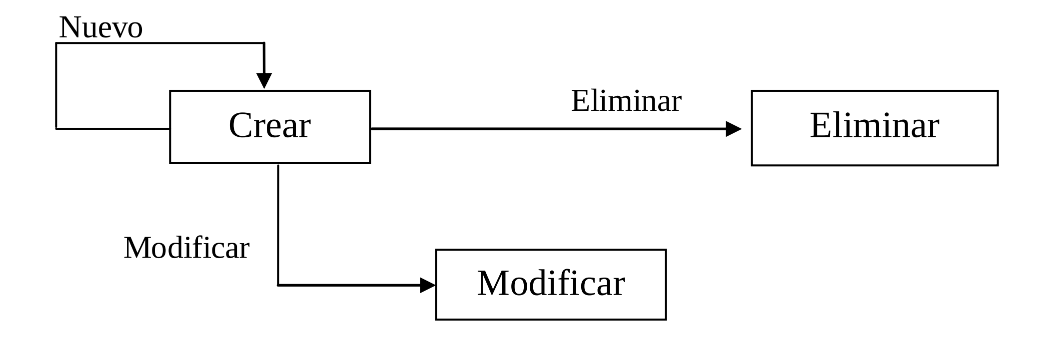
detalles_auditorias_comandos



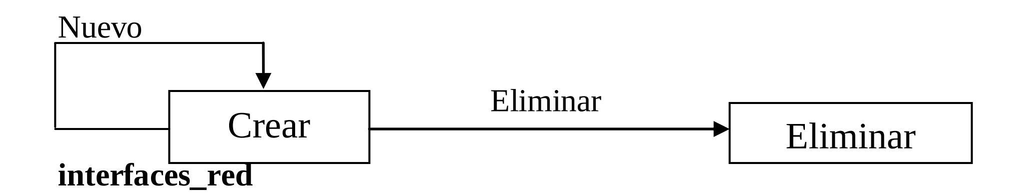
detalles_listas_acceso



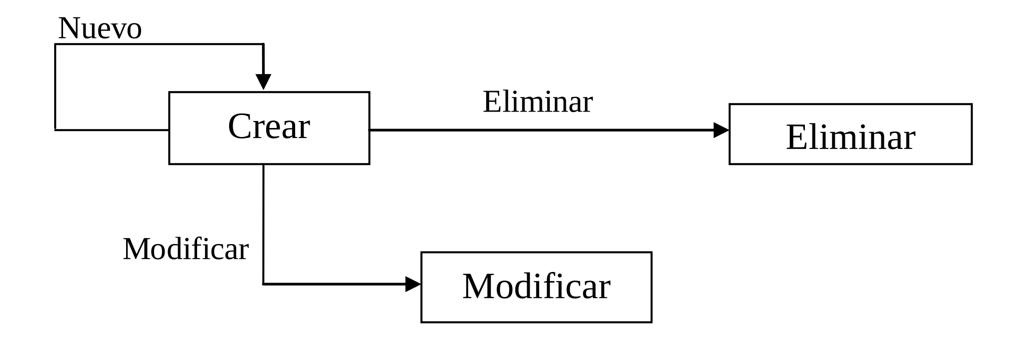
dispositivos_empresas



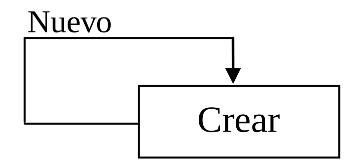
empresas



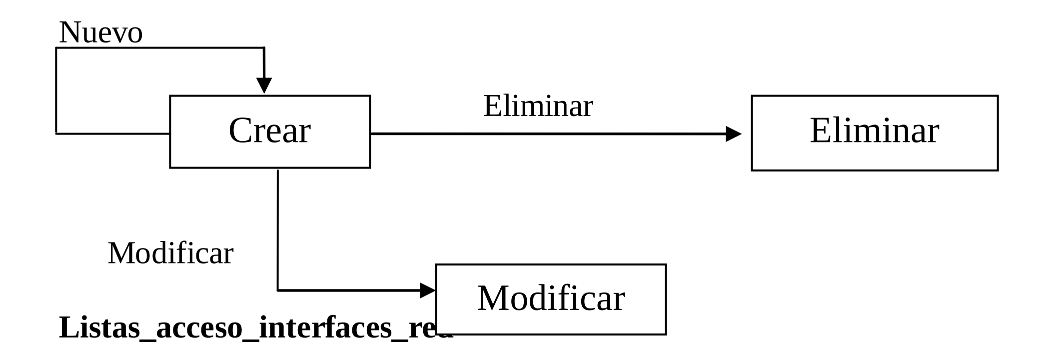
interfaces_red



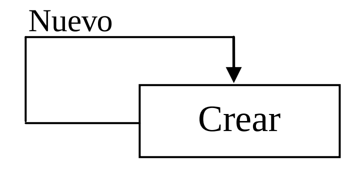
Interfaces_red_politicas_trafico



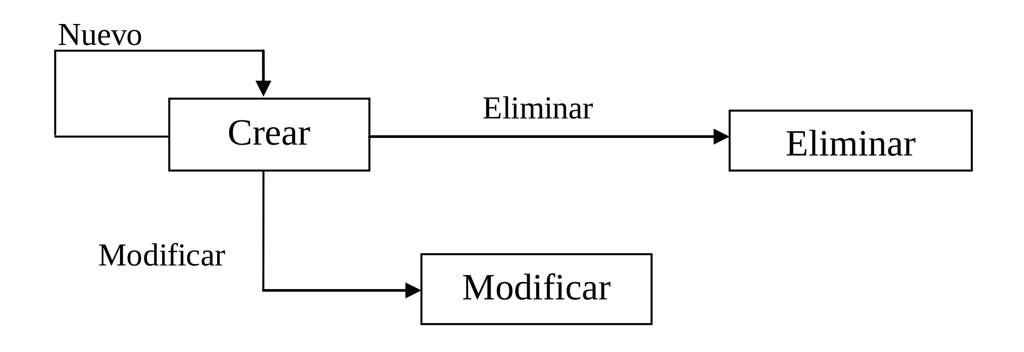
Listas_acceso



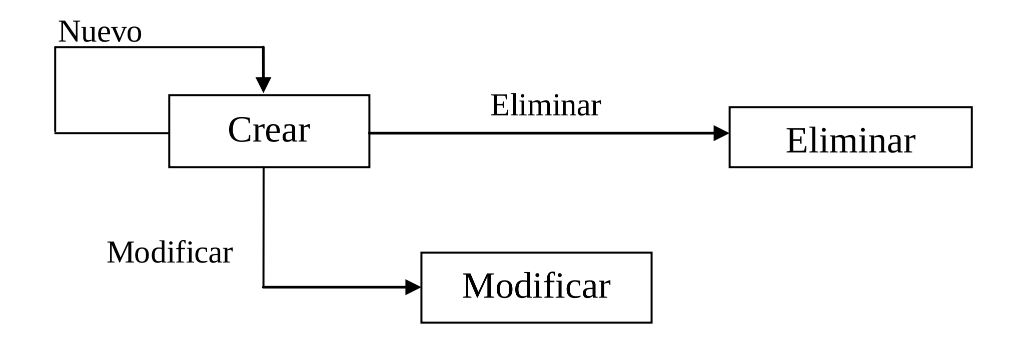
Listas_acceso_interfaces_red



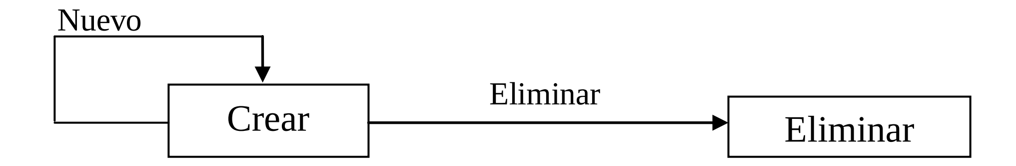
marcas



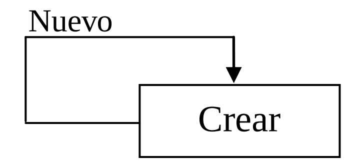
modelos



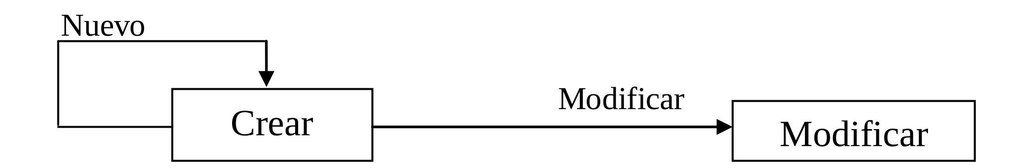
políticas_trafico



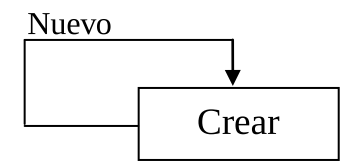
politicas_trafico_modelos



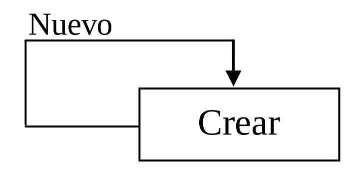
roles



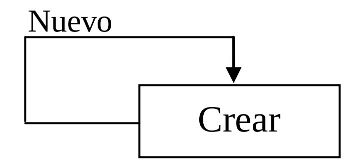
tipos_comandos



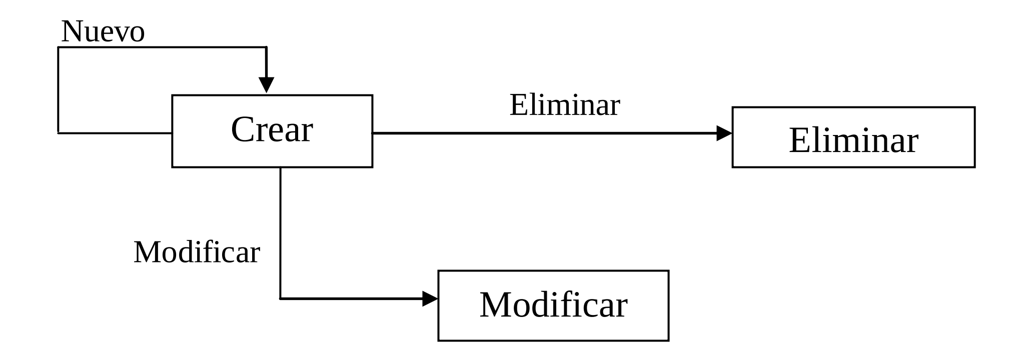
tipos_dispositivos



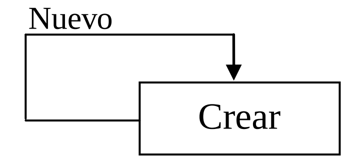
tipos_politicas_traficos



usuario



usuarios_roles



2.5.6. Diccionario de Datos

ID : Auditoria
NOMBRE : Auditoria
ALIAS:
DESCRIPCION: Tabla débil contiene los registros cabeceras de la Auditoria

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Auditoria
LLAVE SECUNDARIA : Id_empresa

OBSERVACION : Aquí se encuentran almacenadas todas las cabeceras de las auditorias.

ID : Comandos
NOMBRE : Comandos
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los comandos ingresados previ o análisis con las normas cisco

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_comando
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados los comandos que le corresponden a las diferentes dispositivos de router y switch cisco

ID : Comandos_modelos
NOMBRE : Comandos_modelos
ALIAS:
DESCRIPCION: Tabla fuerte contiene los enlaces entre la tabla comandos y modelos

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_comandos, Id_modelo
LLAVE SECUNDARIA:

OBSERVACION: Aquí se encuentran almacenadas todas los enlaces que pueden haber entre las tablas comando y modelos

ID : Detalles_Auditorias
NOMBRE : Detalles_Auditorias
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de detalles de las auditorias realizadas

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Detalles_Auditoria
LLAVE SECUNDARIA : Id_Auditroia

OBSERVACION: Aquí se encuentran almacenados los detalles de las auditorías realizadas o sea los dispositivos analizados en esa empresa

ID : Detalle_Listas_Acceso
NOMBRE : Detalle_Listas_Acceso
ALIAS:
DESCRIPCION: Tabla débil contiene los registros detallados de los accesos a los diferentes dispostivos para la empresa

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Detalles_Lista_Acceso
LLAVE SECUNDARIA : Id_Lista_Acceso

OBSERVACION: Aquí se encuentran almacenados los registros de los accesos que tenemos en la empresa

ID : Detalles_Auditoria_Comandos
NOMBRE : Detalles_Auditoria_Comandos
ALIAS:
DESCRIPCION: Tabla débil contiene los registros detallados de los comandos auditados en la empresa

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Detalle_Auditoria_Comandos
LLAVE SECUNDARIA : Id_Detalle_Auditoria, Id_Comando

OBSERVACION : Aquí se encuentran almacenados tofos los detalles de los comandos auditados dependiendo de los dispositivo de cada empresa

ID : Dispositivos_empresas
NOMBRE : Dispositivos_empresas
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de los diferentes dispositivos que la empresa desea que se audite

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Dispositivo_Empresa
LLAVE SECUNDARIA : Id_Empresa, Id_Modelo

OBSERVACION : Aquí se encuentran almacenados todos aquelos dispositivos que las empresas indican para que se les audite

ID : Interfaz_red
NOMBRE : Interfaz_red
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de las interfaces de cada dispositivo que va a ser auditado

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Interfaz
LLAVE SECUNDARIA : Id_Detalles_Auditoria

OBSERVACION: Aquí se encuentran almacenados todas aquellas interfaces de cada dispositivo a ser auditado

ID : Lista_Acceso_Interfaz_Red
NOMBRE : Lista_Acceso_Interfaz_Red
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de las la lista de acceso par ver en que sentido van

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Lista_Acceso_Interfaz_Red
LLAVE SECUNDARIA : Id_Interfaz, Id_Lista_Acceso

OBSERVACION : Aquí se encuentran almacenados los sentidos de las listas de acceso de entrada o de salida

ID : Marca
NOMBRE : Marca
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de las marca de los dispositivos

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Marca
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados los datos de las diferentes marcas de dispositivos

ID : Listas_Acceso
NOMBRE : Listas_Acceso
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de cabecera de los acceso que podemos tener para realizar la auditoria

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Lista_Acceso
LLAVE SECUNDARIA : Id_Detalle_Auditoria

OBSERVACION : Aquí se encuentran almacenados todos los registros que nos indican el acceso a los dispositivos de la empresa

ID : Roles
NOMBRE : Roles
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los diferentes tipos de roles que `puede cumplir un usuario

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Marca
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados los datos de las

ID : Modelo
NOMBRE : Modelo
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de las empresas que nos indican que las auditemos

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Modelo
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados todos los datos de los diferentes modelos de dispositivos

ID : Politicas_Trafico_Modelos
NOMBRE : Politicas_Trafico_Modelos
ALIAS:
DESCRIPCION: Tabla intermedia que contiene registros de relación entre puertos y dispositivos

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Dispositivo_Empresa
LLAVE SECUNDARIA : Id_Empresa, Id_Modelo

OBSERVACION : Aquí se encuentran almacenados todas relaciones entre dispositivos y puertos

ID : Politicas_Trafico
NOMBRE : Politicas_Trafico
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de los puertos a ser auditados

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Dispositivo_Empresa
LLAVE SECUNDARIA : Id_Empresa, Id_Modelo

OBSERVACION : Aquí se encuentran almacenados todos aquellos los datos de los puertos a ser auditados

ID : Empresa
NOMBRE : Empresa
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de las empresas que nos indican que las auditemos

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial
Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Empresa
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados toda la descripción de la empresas que desean que se les audite

ID : Interfaz_Red_Politicas_Trafico
NOMBRE : Interfaz_Red_Politicas_Trafico
ALIAS:
DESCRIPCION: Tabla débil contiene los registros de las buenas prácticas para
realizar las auditorias posteriores

TIPO DE ARCHIVO Manual
Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Interfaz_Red_Política_Tráfico
LLAVE SECUNDARIA : Id_Interfaz, Id_Politica_Trafico

OBSERVACION : Aquí se encuentran almacenados toda las buenas prácticas que recomienda cisco para sus dispoitivos

ID : Tipos_Dispositivos
NOMBRE : Tipos_Dispositivos
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los tipos de dispositivos

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Marca
LLAVE SECUNDARIA :

OBSERVACION: Aquí se encuentran almacenados los datos de los diferentes tipos de dispositivos como por ejemplo switch, router.

ID : Tipos_Comandos
NOMBRE : Tipos_Comandos
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los diferentes comandos que pueda tener un dispositivo

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Modelo
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados todos los nombres de los comandos de los diferentes dispositivos

ID : Usuarios
NOMBRE : Usuarios
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los usuarios que van a manejar el sistema

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Usuario
LLAVE SECUNDARIA :

OBSERVACION: Aquí se encuentran almacenados los datos de cada uno de los usuarios

ID : Tipos_Politicas_Trafico
NOMBRE : Tipos_Politicas_Trafico
ALIAS:
DESCRIPCION: Tabla fuerte contiene los registros de los tipos de políticas que almacenan en el serrvidor

TIPO DE ARCHIVO Manual Computarizado
FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS
LLAVE PRIMARIA : Id_Tipos_Politicas_Trafico
LLAVE SECUNDARIA :

OBSERVACION : Aquí se encuentran almacenados todos los datos correspondientes a las políticas de trafico que se encuentran en el servidor de la empresa

ID : Usuarios_Roles

NOMBRE : Usuarios_Roles

ALIAS:

DESCRIPCION: Tabla débil contiene los registros de los roles asignados a ese usuario

TIPO DE ARCHIVO Manual Computarizado

FORMATO DE ARCHIVO XBD Indexado Secuencial Directo

ESTRUCTURA DE DATOS

LLAVE PRIMARIA : Id_Usuarios_Roles

LLAVE SECUNDARIA : Id_Usuario

OBSERVACION: Aquí se encuentran almacenados los datos correspondientes a roles de ese usuario ya sean estos auditor o administrador

CAPITULO #3

DISEÑO DEL SISTEMA

3.1. Propósito del Diseño.

El sistema está diseñado de forma tal que pueda ser capaz de satisfacer la verificación de las políticas de seguridad de varios tipos de routers y/o switch, debido a que es altamente parametrizable y de esta forma le permite al usuario administrar y manejar información en un alto nivel de eficiencia y eficacia.

3.2 . Menú Principal.

Este menú contiene todas las opciones que se encuentran disponibles en la aplicación, desde aquí el usuario podrá ejecutar las diferentes opciones que listamos a continuación:

- Empresas
- Marcas
- Modelos
- Auditorias
- Comandos
- Políticas de Trafico

- Usuarios
- Cerrar Session

3.2.1. Procesos

Este contiene a las opciones en las cuales se basa las operaciones principales de la aplicación.

3.2.1.1. Conectar A Router y/o Switch

Esta opción le permite al usuario poder conectar la aplicación con el dispositivo Router y/o switch y extraer información necesaria para almacenarla y luego realizar la verificación de la misma.

3.2.1.2. Auditoria De Router y/o Switch

Aquí el usuario realiza la verificación de la información extraída del dispositivo contra las mejores prácticas ingresadas para un modelo determinado de Router.

3.2.1.3. Exportar Datos

Aquí el usuario podrá exportar datos desde la base de datos del cliente hacia archivos pdf.

3.2.2. Consultas

Esta contiene opciones que el usuario podrá utilizar para revisar operaciones realizadas anteriormente.

3.2.2.1. Extracciones Realizadas

Por medio de esta opción el usuario puede revisar las extracciones realizadas anteriormente.

3.2.2.2. Auditorias Anteriores

Por medio de esta opción el usuario puede obtener datos de auditorías realizadas anteriormente.

3.2.3. Reportes

Esta opción contiene algunos tipos de reportes necesarios para el usuario para visualizar los resultados de las diferentes operaciones que se realiza en la aplicación.

3.2.4. Mantenimiento

Contiene todas las opciones de parametrización como por ejemplo: creación de usuarios y permisos.

3.2.4.1. Empresas

En esta opción podrá registrar los datos de las compañías que representen los clientes del sistema.

3.2.4.2. Usuarios

En esta opción maneja tres opciones, las cuales son:

Perfil.- Se asigna descripción a los perfiles del sistema, y se activan o desactivan los mismos.

Asignación de Permisos.- Asigna un perfil al usuario ó configura los permisos que dicho usuario tendrá.

Creación de Usuarios.- Aquí se procede a crear todos los usuarios que accederán al sistema.

3.2.4.3. Cliente

Realiza la creación de los clientes a los cuales se les realizara el control de la auditoria de Router y/o Switch.

3.2.4.4. Router y/o Switch

Esta opción maneja tres opciones las cuales las detallamos aquí:

Creación de Marcas.- Crea las diferentes marcas que se usaran los Router y/o Switch en la aplicación.

Creación de Modelo.- Crea los diferentes modelos que usaran los Router y/o Switch en la aplicación.

Buenas Prácticas Configuración.- Carga las mejores prácticas de configuración y se las asociara a un determinado Router y/o Switch.

Buenas Políticas de Trafico.- Carga las mejores prácticas de tráfico y se las asociara a un determinado Router y/o Switch.

3.2.4.5. Secuencia

Esta opción sirve para asignar diversas secuencias al cliente como de extracción, verificación y xml.

3.2.4.6. Cambio de Clave

Esta opción sirve para que el usuario pueda modificar o reemplazar su clave.

3.2.4.7. Actualiza Clave

Esta opción sirve para restablecer la clave a un usuario que se le ha olvidado.

3.3. Diseño de Interfaz Grafica de Usuario

En esta sección se describirán las pantallas que contiene el sistema.

3.3.1. Conexión a la base de datos

Presentación del Sistema

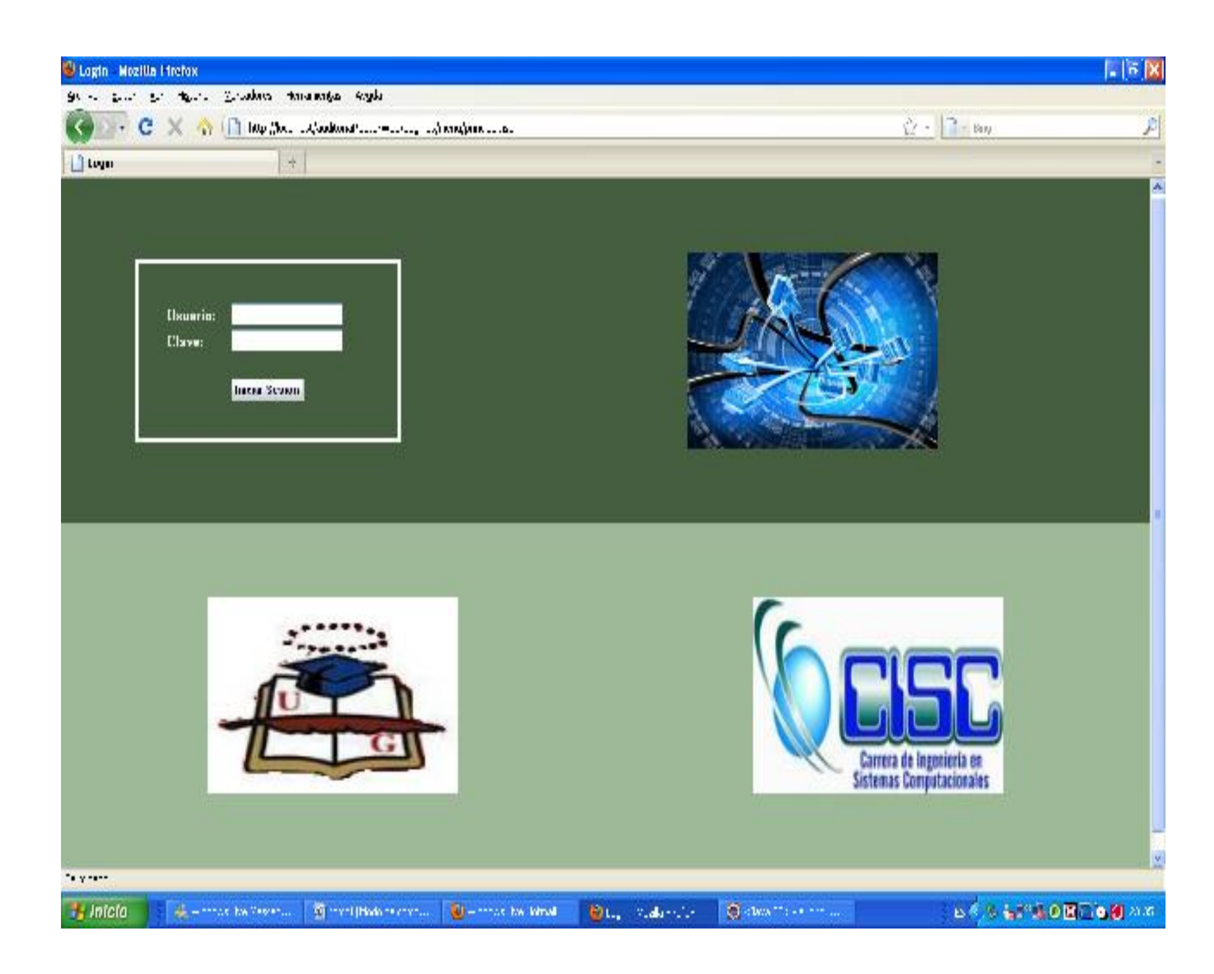


Figura 8 Presentación del Sistema

3.3.2 Menú Principal

Nombre de Campo	**Descripción del campo**
Usuario	Nombre de Usuario
Clave	Contraseña de Usuario
Característica	
Pantalla que permite el inicio de una sesión en el sistema con un usuario determinado	



Figura 9 Menú Principal

Característica
Pantalla que muestra todas las opciones que contiene el servidor

3.3.3 Crear Empresa

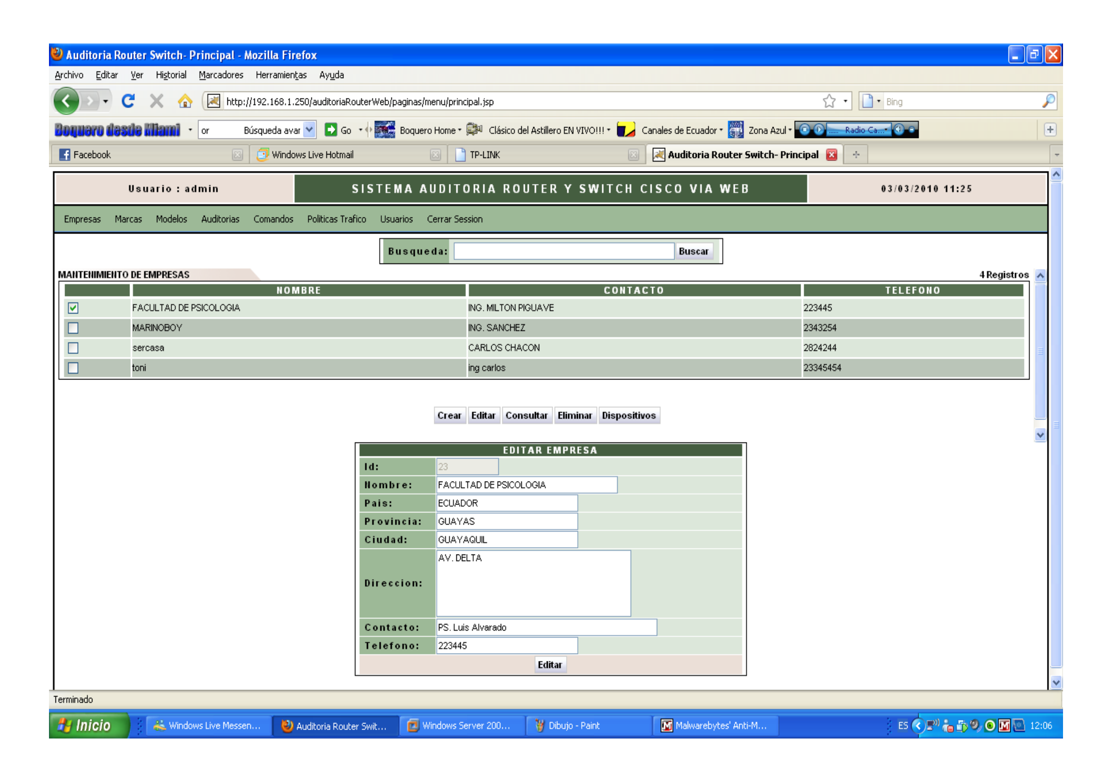


Figura 10 Crear Empresa

Nombre de Campo	**Descripción del campo**
Búsqueda	Busca el nombre de Empresa creada
Nombre	Nombre Empresa
País	País donde reside la Empresa
Provincia	Provincia donde reside la Empresa
Ciudad	Ciudad donde reside la Empresa
Dirección	Lugar de ubicación de la Empresa
Contacto	Jefe o Persona encargada que nos asistirá en la Auditoria
Teléfono	Número telefónico de la empresa a auditar
Característica	
Pantalla que permite la creación de empresa	

3.3.4 Crear Dispositivos

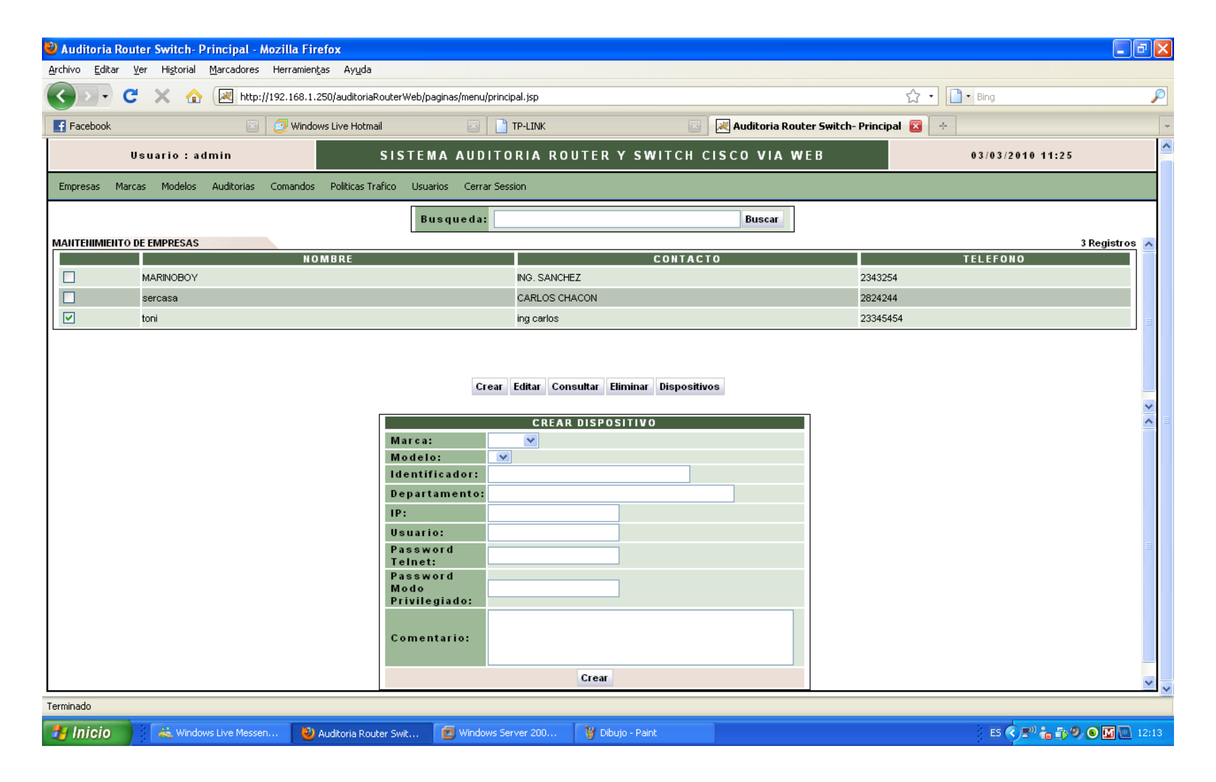


Figura 11 Crear Dispositivos

Nombre de Campo	**Descripción del campo**
Marca	Marca del Dispositivo
Modelo	Modelo del Dispositivo
Identificador	Numero de serie del dispositivo
Departamento	Lugar de ubicación dentro de la empresa del dispositivo
IP	Puerta de enlace para la conexión vía Telnet
Usuario	Usuario creado dentro del dispositivo que pide para la conexión
Password Telnet	Contraseña para conexión via telnet
Password Modo Privilegiado	Contraseña para modo de administrador
Comentario	Algo que nos sirva de relevancia
Característica	
Pantalla que permite la creación del dispositivo para la empresa ya creada	

3.3.5 Crear Marcas

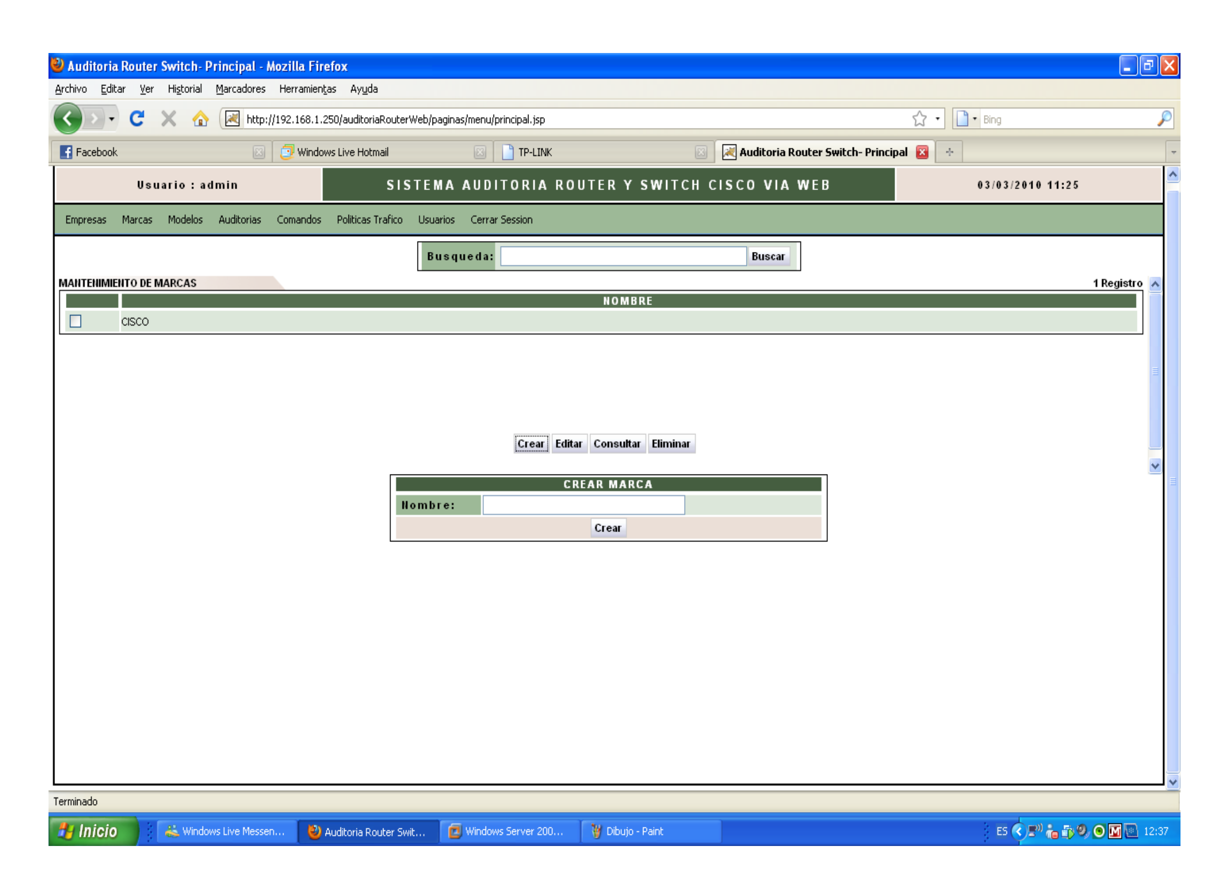


Figura 12 Crear Marcas

Nombre de Campo	**Descripción del campo**
Búsqueda	Busqueda por nombre de marcas
Nombre	Nombre de la Marca de Dispositivo
Característica	
Pantalla que permite la creación de marcas para los dispositivos a utilizar	

3.3.6 Crear Modelos del Router y/o Switch

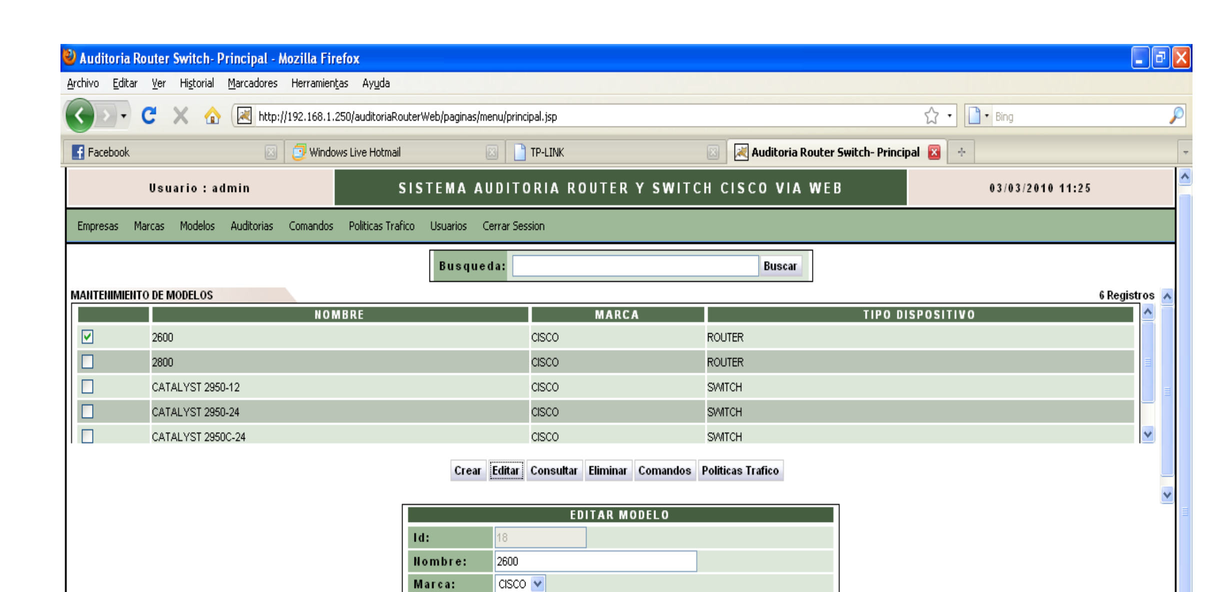


Figura 13 Crear Modelos del Router y/o Switch

Nombre de Campo	**Descripción del campo**
Búsqueda	Busca los modelos ya creados
Id	Identificación dentro de la base datos
Nombre	Nombre del Modelo de Dispositivo
Marca	Marca del dispositivo
Tipo Dispositivo	Puede ser Switch o Router
Característica	
Pantalla nos permite crear modelos Router o Switch	

3.3.7 Crear Auditoria

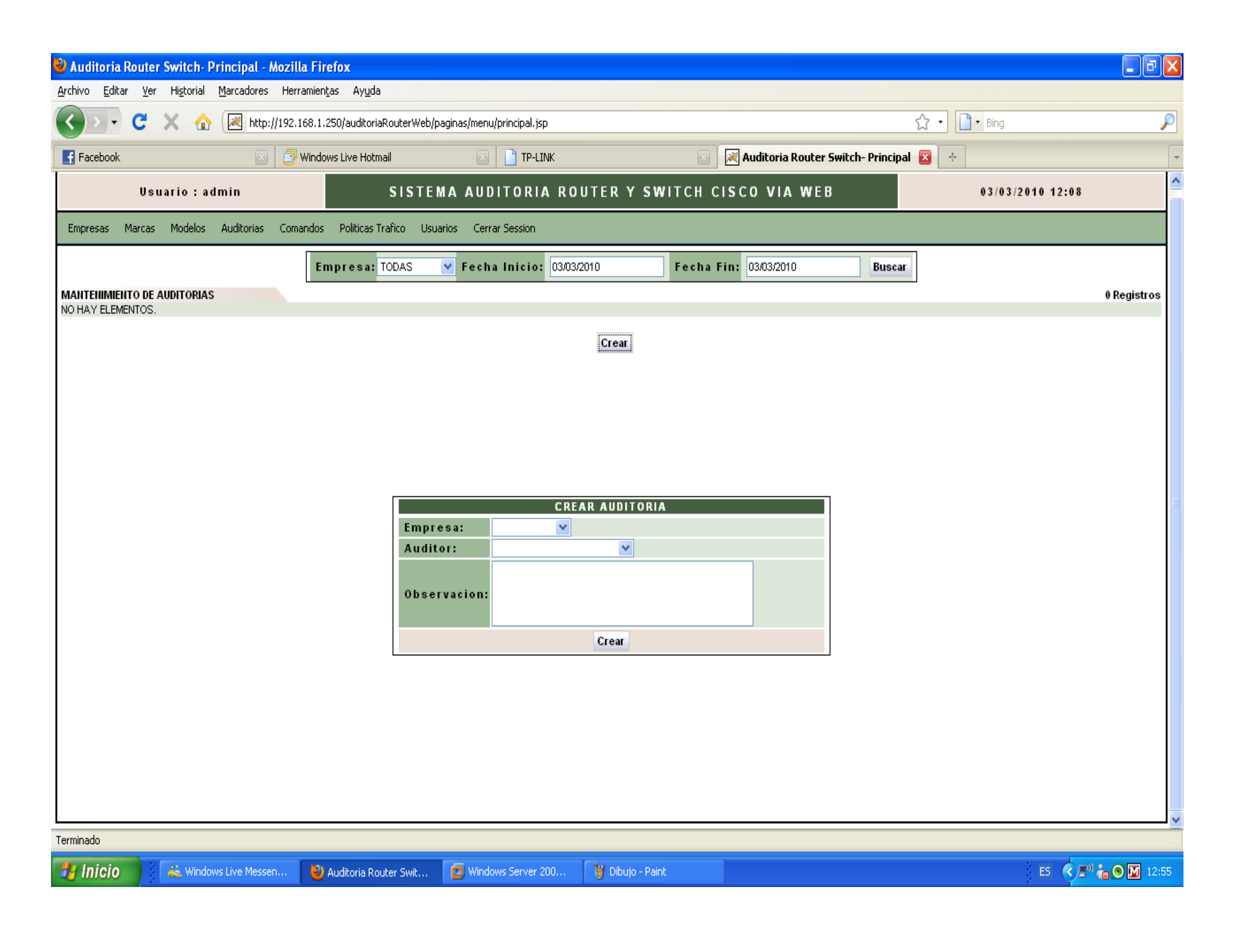


Figura 14 Crear Auditoria

Nombre de Campo	**Descripción del campo**
Empresa	Busca las empresas que han sido auditadas
Fecha Inicio	Rango de fecha de inicio
Fecha Fin	Rango de fecha de fin
Empresa	Nombre de la empresa
Auditor	Persona que va a realizar dicha auditoria
Observaciones	Observación dentro de la creación de la auditoria
Característica	
Pantalla nos permite crear auditorias y hacer búsquedas de las mismas	

3.3.8 Crea y Añade Dispositivo

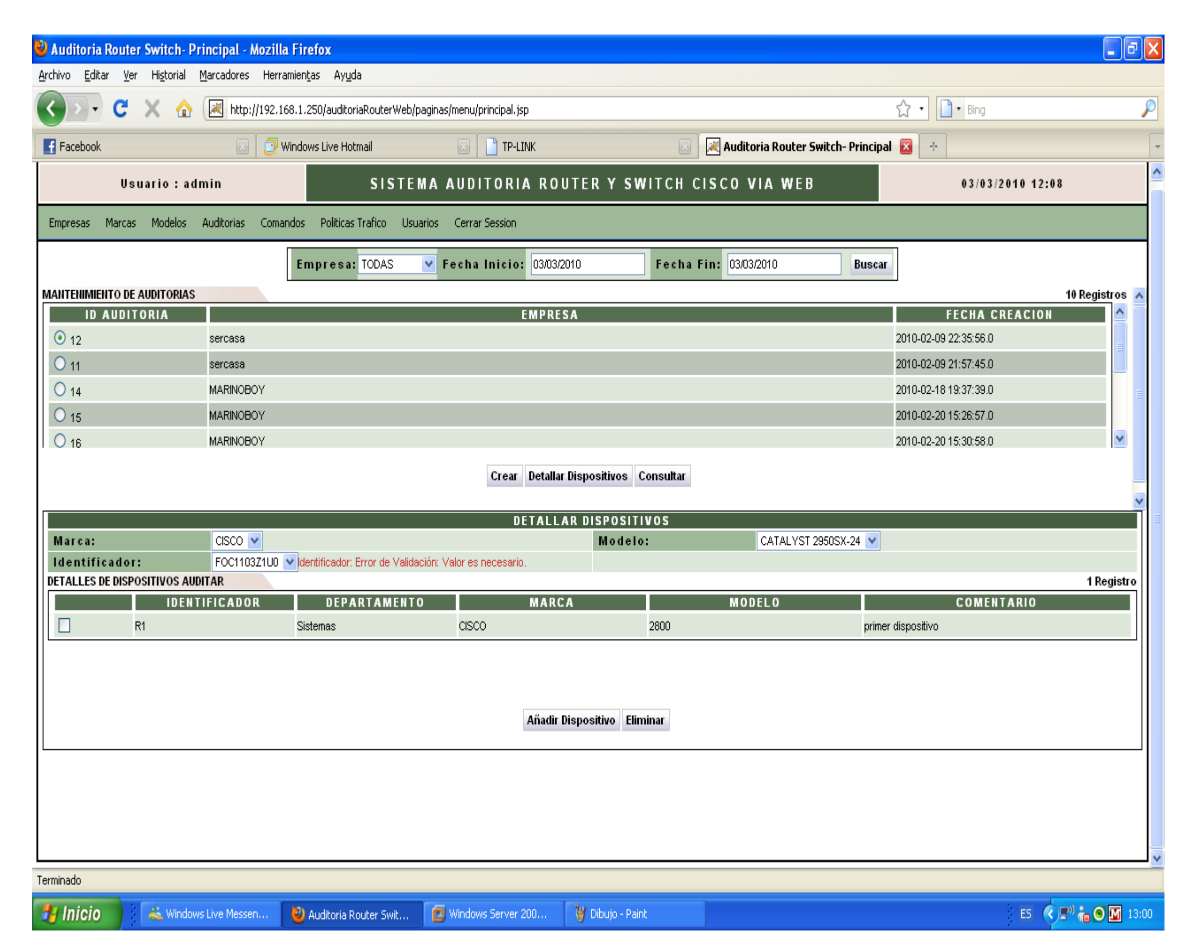


Figura 15 Crea y Añade Dispositivo

Nombre de Campo	**Descripción del campo**
Marca	Marca creada para añadir dispositivo
Modelo	Modelos Creado para la empresa en dicha auditoria
Identificador	Numero de seria para la empresa en dicha auditoria
Característica	
Pantalla nos permite crear y añadir dispositivos para la empresa que va a ser auditada	

3.3.9 Crear Comandos

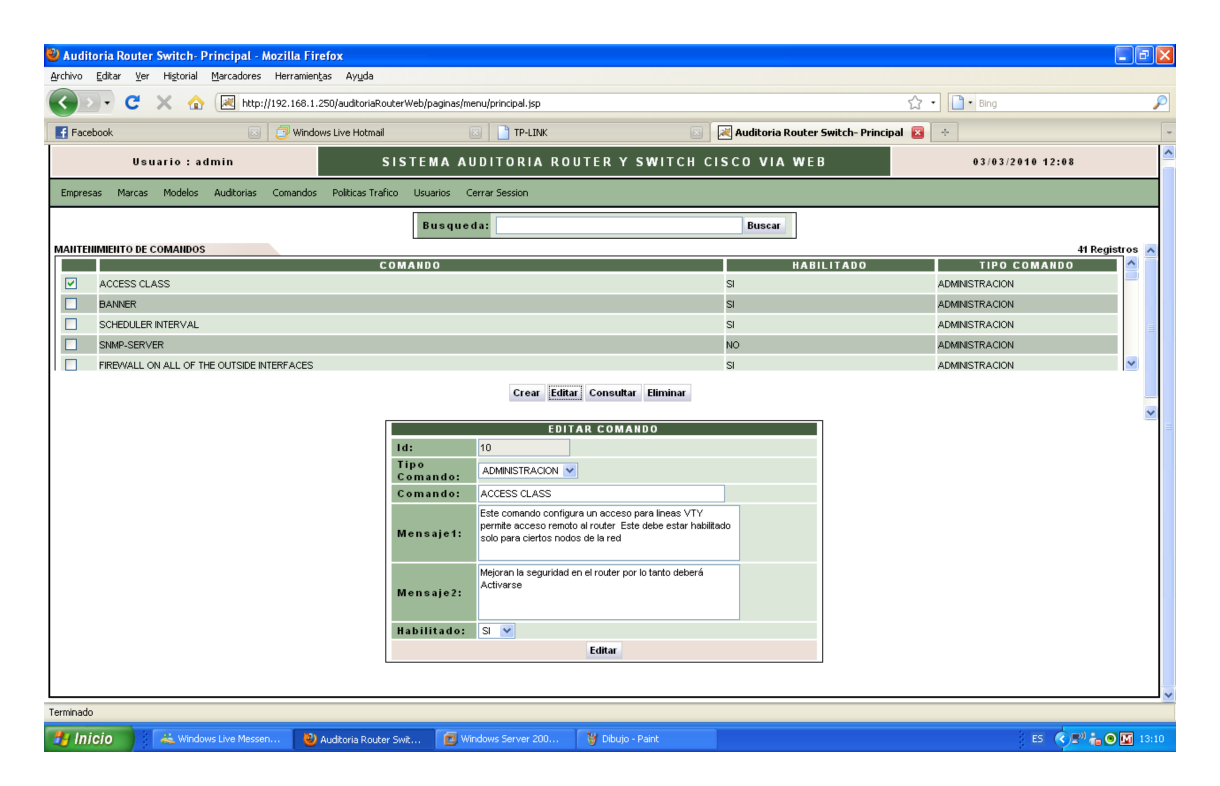


Figura 16 Crea Comandos

Nombre de Campo	**Descripción del campo**
Búsqueda	Busca comandos creados
Id	Secuencial dentro de la base de datos
Tipo Comando	Clasificación de los comando
Comando	Nombre de Comando
Mensaje 1	Descripción de Comando
Mensaje 2	Recomendación de Comando
Habilitado	Comando debe estar configurado dentro del dispositivo
Característica	
Pantalla nos permite crea comandos para los dispositivo	

3.3.10 Crea Protocolos y Puertos

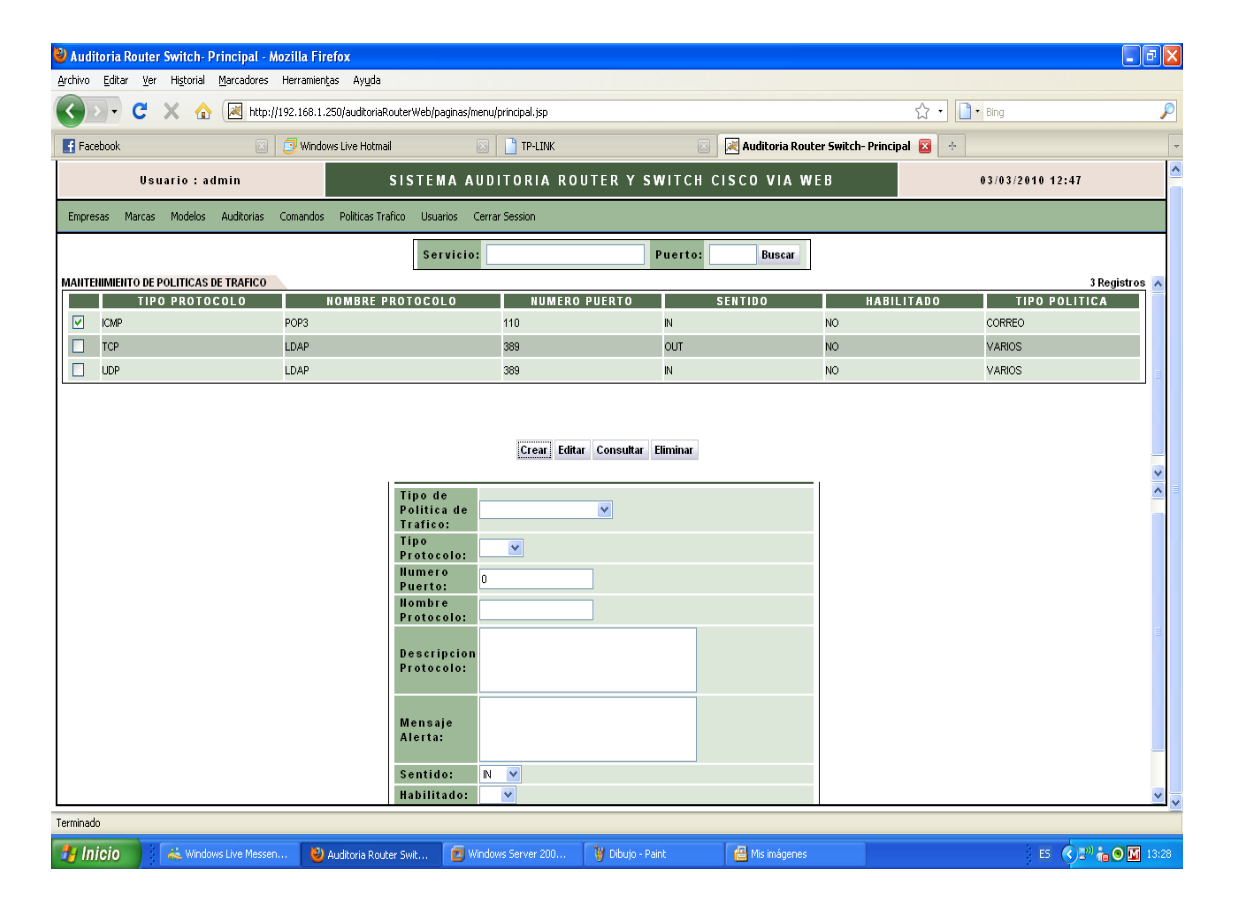


Figura 17 Crea Protocolos y Puertos

Nombre de Campo	**Descripción del campo**
Servicio	Busca tipo de protocolo creados , va de la mano con Puerto
Puerto	Busca numero de puerto creados, va de la mano con Servicio
Tipo Politica de Trafico	Clasificación del tipo de política
Tipo de Protocolo	Clasificación de Protocolo
Numero de Puerto	Numero de puerto que se utiliza
Nombre de Protocolo	Nombre de Protocolo
Descripcion Protocolo	Descripción de Protocolo a ser creado
Mensaje Alerta	Recomendación del Protocolo
Sentido	Sentido donde se aplica las lista de acceso
Habilitado	Se aplica en la auditoria
Característica	
Pantalla nos permite crea protocolo y puerto que serán verificados dentro de las auditorias	

3.3.11 Crear Usuarios

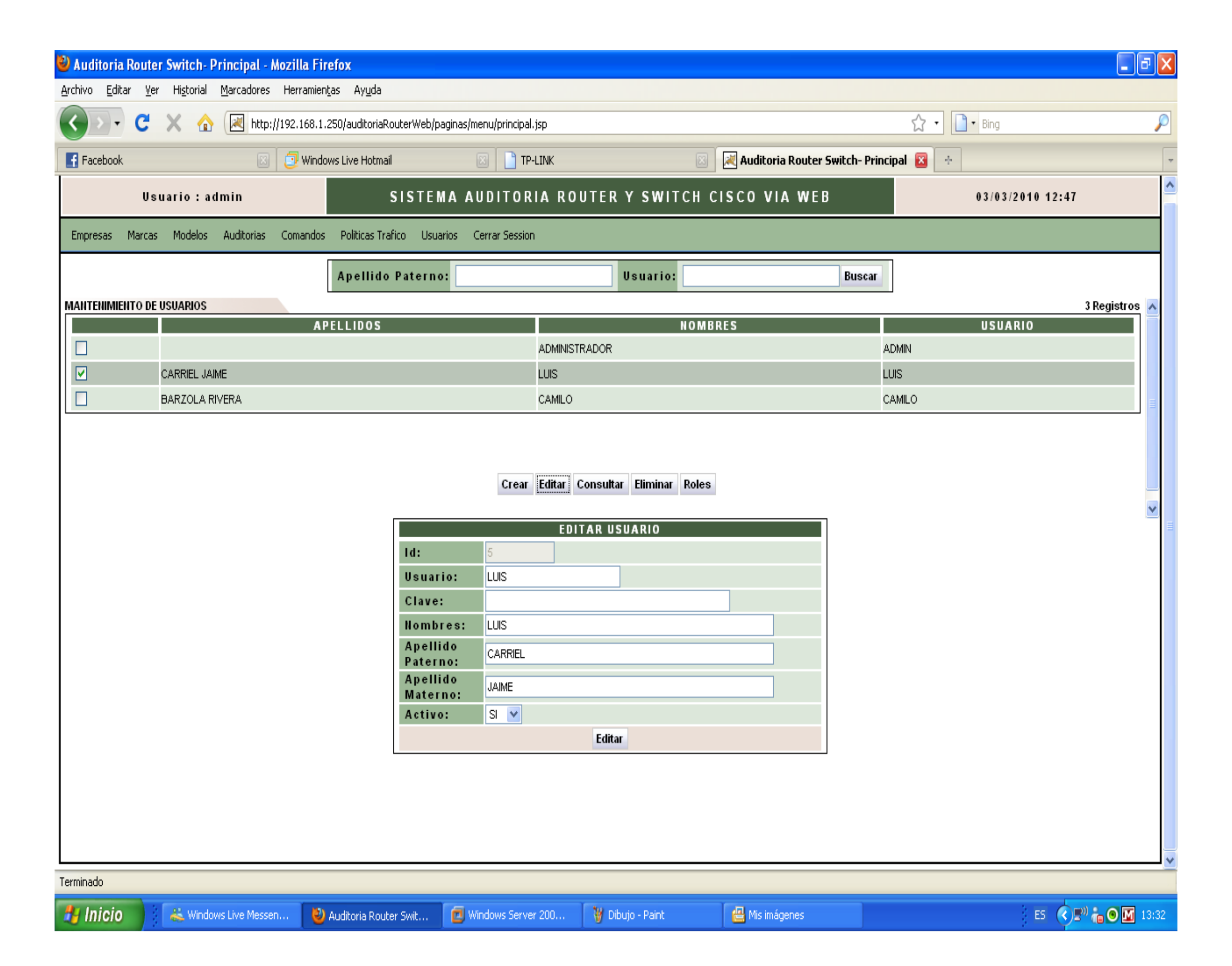
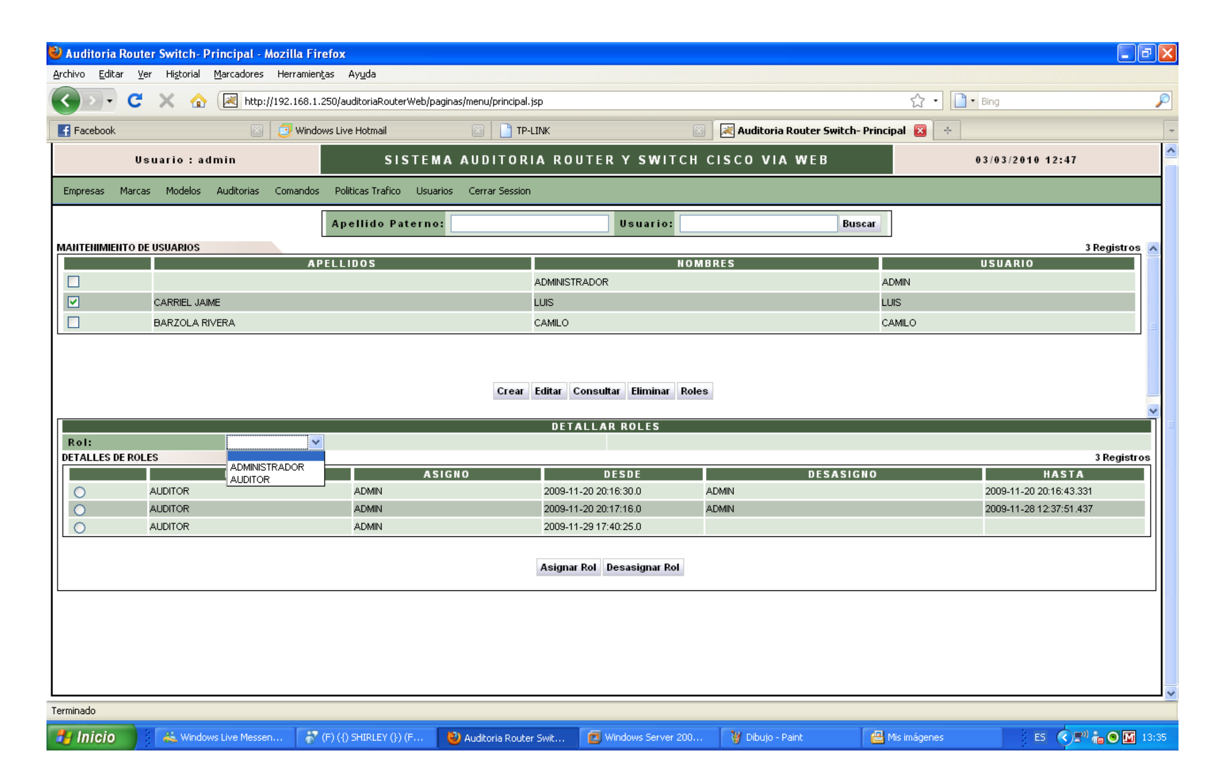


Figura 18 Crear Usuarios

Nombre de Campo	**Descripción del campo**
Apellido Paterno	Busca por apellido de paterno de usuario
Usuario	Busca Nombre de Usuario
Id	Secuencia en base de datos
Clave	Contraseña de Usuario
Nombre	Nombre de Usuario
Apellido Paterno	Apellido Paterno de Usuario
Apellido Materno	Apellido Materno de Usuario
Activo	Estado de Usuario
Característica	
Pantalla nos permite crear usuarios que van a se administradores o auditores del servidor	

3.3.12 Crear Roles



Figuras 19 Crear Roles

Nombre de Campo	Descripción del campo
Rol	Asignación de usuario que le da el administrador
Característica	
Pantalla que permite asignar roles por parte del administrador del servidor	

CAPITULO #4

4 DESARROLLO DEL SOFTWARE

4.1 CODIFICACION DE LOS PRINCIPALES COMPONENTE

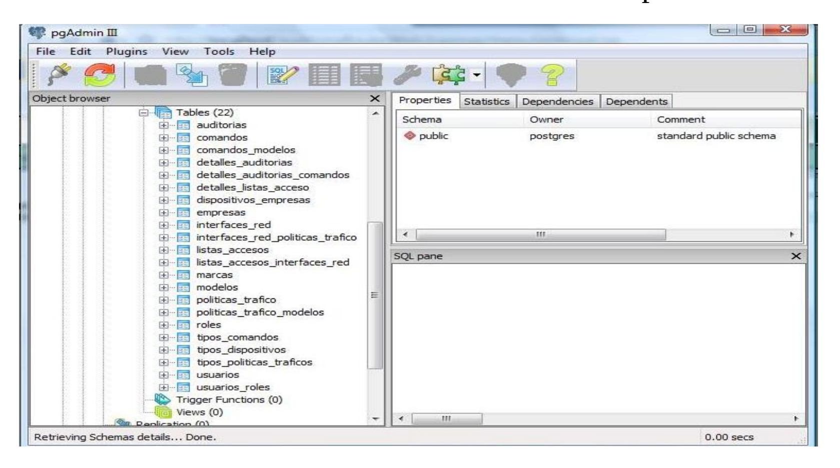
4.1.1 Proceso de Conexión a Router o Switch

Este proceso se utilizara para extraer en forma sensitiva la configuración del router o switch cisco eta información puede ser obtenida de dos formas que son las siguientes:

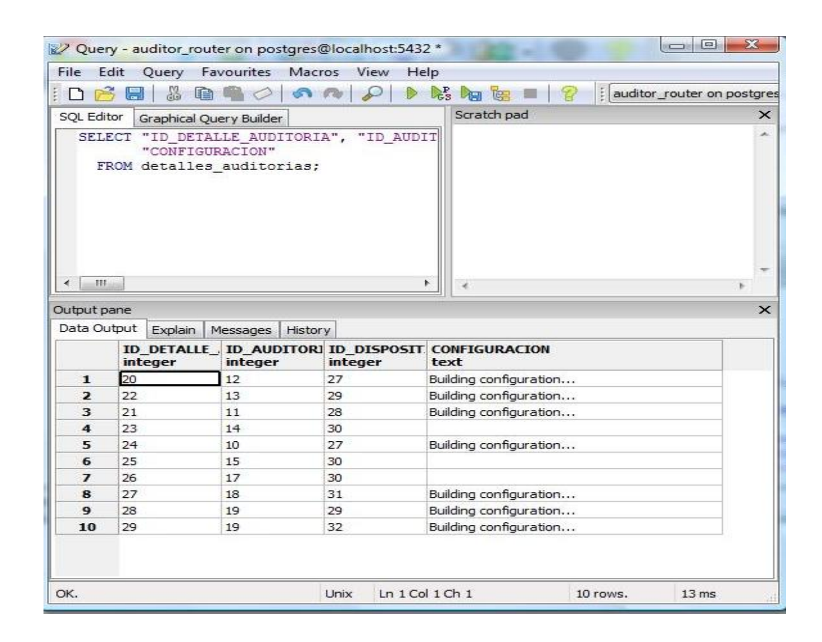
Se debe de tener conectar el computador y los dispositivos a ser auditados en este caso es un router o switch y el usuario deberá de digitar su nombre y clave de acceso para realizar la extracción del dispositivo en este caso les aparcera las empresas con sus auditorías asignadas y se escoge una de ellas y se realiza la extracción se la realiza vía telnet.

Permitir importar un el contenido de un archivo de texto puesto en una carpeta determinada.

En esta tabla se almacena las extracciones realizadas a los dispositivos



Figuras 20 Base de datos Postgre SQL 8.4



Figuras 21 Datos Extraídos del Router o Switch Cisco

4.1.2 Proceso de Obtención de Información

Este proceso permite generar reportes primeramente debemos de escoger la auditoria a ser presentad y este aparecerá impresa por pantalla o impreso

4.1.3 Proceso de Verificación

Este proceso nos permite comparar la configuración extraída de los dispositivos con las buenas prácticas

4.1.4 Proceso de Asignación

Nos permite asignar a los roles de los usuarios y como también quien irá a realizar la respectiva auditoria en la empresa que nos contrato

4.2 DESARROLLO DE PRUEBAS E IMPLEMENTACION

4.2.1 CREACION DE LA BASE DEDATOS

Para la creación de la base de datos se ha elegido, la base de datos PostgreSql versión 8.4 y sabiendo que los objetos de conexión y manipulación de la base de datos están realizados bajo lenguaje java

4.2.2 Pruebas del Sistema

4.2.2.1 Prueba de Aplicación Ensamblada

Hemos venido realizando pruebas para ver cuales serian los posibles errores ya que nuestro sistema se encuentra enlazados todos los módulos

4.2.2.2 Prueba de Aplicación con Varios Usuarios

SE ha realizado prueba con varios usuarios tanto sean estos administradores como auditores con diferentes router y switch cisco, vale recalcar que nuestro sistema es de fácil manejo

4. 3 IMPLEMENTACION DEL SISTEMA

4.3.1 Componentes del Software

1) Lenguaje de programación java con sus componentes Galileo y Ganimede

2) Base de datos PostgreSql 8.4

3) Utilización de un servidor web Apache Tomcat 6.0

4) Reporteria IText de java

4.3.2 Componentes del Hardware

1 Pc. Pentium Dual Core con disco duro de 80 GB y 512 Mb de memoria Ram esta puede tener sistema operativo WINDOWS XP

CAPÍTULO #5

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Recomendaciones.

En el ambiente del auditor el tomara el usuario, clave de los dispositivo auditados y la dirección ip del servidor donde está instalado el ambiente de administrador, tener en cuenta que si falla la extracción por medio de telnet de algún dispositivo auditado, es porque no han ingresado la claves correctamente en el servidor y los administradores de sistemas de las empresas auditadas tienen que dar las claves que se utilizan en los dispositivos para que la extracción se exitosa o si no se puede utiliza mediante un archivo extraído mediante tftp donde la llevaremos a nuestra carpeta creada que se encuentra en el servidor donde se encuentra asilado el ambiente del administrador y tener muy en cuenta que de esta tenemos que haber guardado nuestro archivo de la siguiente manera que es el código de la auditoria seguido con el sub-guión mas el número de serie del dispositivo con estos parámetros el auditor pueden seguir con los pasos de la auditoria ya que podemos procesarla y emitir nuestro reporte del o los dispositivos que tengo las empresas.

5.2 Conclusiones.

Para la implementación del sistema se realizo varios pasos que establecieron el desarrollo del mismo, como son el análisis, el diseño, la codificación, y termino con las pruebas y puesta en marcha del sistema.

Es un producto estandarizado y sujeto a cambios para nuevos requerimientos del usuario.

El administrador podrá crear privilegios dependiendo el tipo de usuario que lo ejecute por ese motivo se podrán crear nuevos usuarios, empresas, cambiar claves, modificar usuarios, modelos y marcas de router además el sistema utiliza fechas para el acceso al sistema y saber quién realiza la auditoria.

ANEXOS

Entrevista para Establecer las Buenas Políticas de la Auditoria

18. ¿Cuál es su nombre completo?

    Ing. Carlos Pinos

19. ¿Cuánta Experiencia tiene Administrando Routers y/o Switch?

    14 años

20. ¿Qué cursos a realizado para administración de Routers y/o Switch?

    Introducción al diseño de redes

    Operación de un router y/o

    Switch

21. ¿Cuántos equipos de hardware de comunicación hay en la empresa?

    Dentro de la empresa hay 50 routers y/o switch

22. ¿De qué marca y modelo son sus equipos?

    Cisco 1700, Cisco 2600, Cisco 2700, Dell, Dlink, Linksys y Sonicwall.

23. ¿Alguien le recomendó los equipos que utiliza su empresa, si es así nos podría usted decir bajo que parámetros lo hizo?

    Se cuenta con la asesoría técnica de una empresa llamada electrónica digital que nos da asesoría en base de equipos, además de eso se hace un previo análisis de los posibles equipos a adquirir en cuanto costo y beneficio.

24. ¿Qué tipo de software utilizan sus equipos?

    Los equipos utilizan su propio software

25. ¿Que tipos de políticas de seguridad utiliza su empresa para la extracción de datos específicamente de un equipo router y/o switch? ¿Cuáles son los objetivos claves de estas políticas?

    Solo se extrae información del router y/o switch cuando se ha detectado algún problema.

26. ¿Cuales son los parámetros para establecer las políticas de seguridad? Y Porque? ¿Cuáles son los riesgos que se intentan mitigar con estas políticas?

    Como política de seguridad bloquear acceso a internet de manera total y se asigna a cada usuario los permisos de accesos que están relacionados a su

trabajo, los riesgos que se quieren evitar son manejo indebido de información y ataques de hackers.

27. ¿Existe alguna relación entre la topología de red con la manera que usted maneja la seguridad de su empresa?

    Si existe relación ya que la red esta segmentada y dependiendo de este diseño se utiliza el router y/o switch.

28. ¿Cuales son los procedimientos que usted realiza para evaluar la seguridad de sus equipos de comunicación?

    Cada vez que se maneja un equipo de comunicación se debe hacer pruebas de verificación analizando si el equipo cumple con las especificaciones que dice brindar.

29. En caso de problemas, ¿Cuales han sido las medidas o pasos tomados para mitigar los riesgos con los equipos de comunicación (routers y/o switch)?

    Cuando se tiene un problema en la red , lo que se hace es que se corta el internet, se bloquea toda la información encontrada, se da paso a una auditoria activando log detallados para saber lo que hacen los usuarios, luego de esto se analiza que maquina causo el problema, y se lo arregla

30. Estadísticamente si es posible, podría determinar el comportamiento de la seguridad de sus equipos de comunicación? Siempre seguros, % de problemas por año, tipos de problemas, recurrencia de problemas por tipo, etc.

    Problemas muy pocos, ya que la seguridad dentro de la empresa es de un 90 %, como problema recurrente son los virus ya que cada dice nace uno nuevo, no hay intrusos en la red

31. ¿De qué forma usted está seguro de que las políticas de seguridad de los equipos son actualizadas convenientemente y usted las conoce?

    Tener documentada la información acerca de las políticas aplicadas, actualizar la información acerca de los empleados de la empresa para revisar los accesos al sistema o red.

BIBLIOGRAFÍA

http://www.cisco.com: Página Oficial de Cisco

http://www.microsoft.com/windowsserver2003/default.msp : Página Oficial de Windows Server 2003

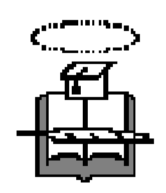
http://tomcat.apache.org : Pagina Oficial de Tomcat

http://commons.apache.org/net : Pagina Oficial para Librerías de telnet

http://itextpdf.com : Pagina para Crear reportes en Java

Guía del usuario de Cisco Router and Security Device Manager

Software de Pruebas: Cisco Packet Tracer



UNIVERSIDAD DE GUAYAQUIL

Facultad de Ciencias Matemáticas y Físicas

Carrera de Ingeniería en Sistemas Computacionales

"SISTEMA DE AUDITORÍA DE SEGURIDADES DE ROUTER Y SWITCH CISCO VIA WEB"

PROYECTO DE GRADO

CURSO DE GRADUACIÓN

Previo a la Obtención del Título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

Autores:

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

GUAYAQUIL-ECUADOR

Año: 2010

AGRADECIMIENTO

Damos gracias a Dios por habernos permitido alcanzar la meta profesional que nos propusimos.

A nuestros padres que con amor y sacrificio acompañaron cada paso de nuestras vidas estudiantiles y nos supieron conducir por el camino de los grandes ideales.

A nuestros amigos mas cercanos que nos dieron todo su apoyo de manera incondicional.

A los profesores y compañeros que han iluminado y compartido cada uno de los rincones de nuestras etapas de estudios.

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

DEDICATORIA

Damos gracias a Dios por habernos permitido alcanzar la meta profesional que nos propusimos.

A nuestros padres que con amor y sacrificio acompañaron cada paso de nuestras vidas estudiantiles y nos supieron conducir por el camino de los grandes ideales.

A nuestros amigos mas cercanos que nos dieron todo su apoyo de manera incondicional.

A los profesores y compañeros que han iluminado y compartido cada uno de los rincones de nuestras etapas de estudios.

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

TRIBUNAL DE GRADUACIÓN

_________________________ _________________________

Presidente **1er. Vocal**

_________________________ _________________________

2do. Vocal **Secretario**

DECLARACIÓN EXPRESA

"La autoría de la tesis de grado corresponde exclusivamente a los suscritos, perteneciendo a la Universidad de Guayaquil los derechos que generen la aplicación de la misma"

(Reglamento de Graduación de la Carrera de Ingeniería en sistemas Computacionales, Art. 26)

Barzola Rivera Camilo Hernán

Carriel Jaime Luis Alberto

Chacón Terán Carlos Alberto

RESUMEN

A continuación detallaremos lo que es una auditoria de router y/o switch, este puede ser de software o hardware y es aquel que comprueba la información procedente de Internet o una red y a continuación, deniega o permite el paso de ésta al equipo, en función de la configuración del dispositivo. De este modo, me ayuda a impedir que los hackers y software malintencionado obtengan acceso al mismo.

La seguridad ha sido el tema principal a tratar cuando una organización desea conectar su red privada al Internet. Sin tomar en cuenta el tipo de negocios, se ha incrementado el número de usuarios de redes privadas por la demanda del acceso a los servicios de Internet, ya que es un medio que nos permite estar comunicados tanto externamente como internamente.

Se desarrollara un sistema que haga posible la comparación de datos extraídos de un router y/o switch contra las buenas políticas de seguridad, que los administradores de red han establecido de acuerdo a las necesidades de la organización.

Se ha determinado las buenas políticas mediante entrevistas a algunos expertos del área. Debido a que los administradores de red tienen que desarrollar todo lo

concerniente a la seguridad de sus sistemas, ya que se expone la organización privada de sus datos así como la infraestructura de su red a los Expertos de Internet.

El sistema a desarrollar tendrá la capacidad de analizar y almacenar los datos extraídos en una base de datos, manejar perfiles de administrador y auditor con sus debidos permisos o restricciones.

Generara reportes y los resultados de las auditorias de router y/o switch.

Lo más importante del sistema a desarrollar es que podrá dar sugerencias al auditor acerca de las vulnerabilidades del router emitiendo el respectivo reporte y recomendación para mitigar dichas vulnerabilidades. Cabe recalcar que la decisión la toma los administradores de las areas de sistemas de las empresas auditadas.

.

INDICE

AGRADECIMIENTO II

DEDICATORIA III

TRIBUNAL DE GRADUACIÓN IV

DECLARACIÓN EXPRESA V

RESUMEN VI

INDICE VIII

INDICE DE FIGURAS XI

INDICE DE CUADROS XIV

1. **MANUAL TECNICO**...1.

1.1. Creación de Tablas De Base de Datos...1

1.2. Procesos Principales. 22

1.2.1 Recuperador Configuración Dispositivo……............................22.

1.2.2 Panel Resultado Procesamiento…..27.

1.2.3.AuditoriaBO 34

1.2.4. Cargador Archivos..49

2. **MANUAL DE USUARIO**..53

2.1. Introducción. ....53.

2.2 La Auditoria…………………………………………………..53

2.3 La Auditoria Sistematizada…………………………………….54

2.4 Hardware y software requeridos……………………………………………54
2.5 Contenido del manual……………………………………………………54
2.6 Primera instalación……………………………………………………..55
2.7 Limite usuario…………………………………………………………..55
2.8 Acceso…………………………………………………………………..55.
2.8.1 Sistemas de Acceso………………………………………………. 56
2.8.2 Niveles de Acceso…………………………………………………56
2.9 Procesos…………………………………………………………………59
2.9.1 Conectar a Router y/o Switch Cisco………………………………59
2.9.2 Extracción de datos…………………………………………………...61
2.9.3 Importación datos…………………………………………………..62
2.9.4 Auditoria de router…………………………………………………62
2.9.5 Salir………………………………………………….………………63
3.0 Mantenimiento……………………………………………………………63
3.0.1 Empresa……………………………………………………………… 63
3.0.2 Marca………………………………………………………………..68
3.0.3 Modelos………………………………………………………………72
3.0.4 Comandos y Políticas de Dispositivo……..…………………………76
3.0.5 Auditorias……………………………………………………………78
3.0.6 Comandos……………………………………………………………..81
3.0.7 Políticas de Tráfico…………………………………………………84
3.0.8 Usuario…………………………………………………….. …..88

3.1 Reporte……………………………………………………………………94

3.1.1 Reporte de Auditorías realizadas…………………………………94

Bibliografía………………………………………………………………….…97

INDICE DE FIGURAS

Figura A Pantalla Inicio 56

Figura B Pantalla del Administrador 57

Figura C Pantalla del Auditor 58

Figura D1 Pantalla Conectarse a Router y/o Switch 59

Figura D2 Pantalla Conectarse a Router y/o Switch 60

Figura E Extracción de Datos 61

Figura F Auditoria 62

Figura G Crear Empresas 63

Figura H Editar Empresas 64

Figura I Consultar Empresas 65

Figura J Eliminar Empresas 66

Figura K Dispositivos 67

Figura L Crear Marcas 68

Figura M Consultar Marcas 69

Figura N Editar Marcas 70

Figura O Elminar Marcas 71

Figura P Crear Modelos 72

Figura Q Editar Modelos 73

Figura R Consultar Modelos 74

Figura S Eliminar Modelos 75

Figura T Comandos de Dispositivo 76

Figura U Políticas del dispositivo 77

Figura V Crear Auditoria 78

Figura W Consultar Auditoria 79

Figura X Detalle Dispositivo 80

Figura Y Crear Comandos 81

Figura Z Editar Comandos 82

Figura AA Eliminar Comandos 83

Figura AB Crear Políticas 84

Figura AC Editar Políticas 85

Figura AD Consultar Políticas de Trafico 86

Figura AE Eliminar Políticas de Trafico 87

Figura AF Crear Usuarios 88

Figura AG Editar Usuarios 89

Figura AH Consulta Usuarios 90

Figura AI Eliminar Usuarios 91

Figura AJ Roles a los Usuarios 92

Figura AK Salir de Sesión 93

Figura AL Reporte de Auditorías realizadas 94

INDICE DE CUADROS

Cuadro A Creación Tabla Auditorias 2

Cuadro B Creación Tabla Comandos 3

Cuadro C Creación Tabla Comandos_modelos 3

Cuadro D Creación Tabla Detalles_auditorias 4

Cuadro E Creación Tabla Detalles_auditorias_comandos 6

Cuadro F Creación Tabla Detalles_listas_acceso 7

Cuadro G Creación Tabla Dispositivos_empresas 8

Cuadro H Creación Tabla Empresas 9

Cuadro I Creación Tabla Interfaces_red 10

Cuadro J Creación Tabla Interfaces_red_politicas_trafico 11

Cuadro K Creación Tabla Listas_accesos 12

Cuadro L Creación Tabla Listas_acceso_interfaces_red 13

Cuadro M Creación Tabla Marcas 14

Cuadro N Creación Tabla Modelos 15

Cuadro O Creación Tabla Políticas_trafico 16

Cuadro P Creación Tabla Políticas_trafico_modelos 17

Cuadro Q Creación Tabla Roles 18

Cuadro R Creación Tabla Tipos_comandos 19

Cuadro S Creación Tabla Tipos_dispositivos 20

Cuadro T Creación Tabla Tipos_politicas_traficos 21

Cuadro U Creación Tabla Usuario 22

Cuadro V Creación Tabla Usuarios_roles 23

Cuadro W Código RecuperadorConfiguracionDispositivo.java 27

Cuadro X Código de PanelResultadoProcesamiento.java 34

Cuadro Y Código de AuditoriaBO.java 49

Cuadro Z Código de CargadorArchivos.java 52

CAPÍTULO 1

1. **MANUAL TECNICO**

1.1 Creación de Tablas De Base de Datos

Auditorias.- En esta tabla se almacenan los datos de cuando se crea una auditoria

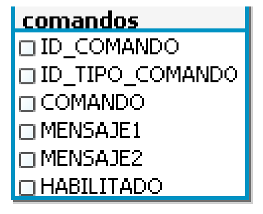
auditorias
ID_AUDITORIA
ID_EMPRESA
FECHA_CREACION
FECHA_AUDITORIA
OBSERVACION_CREACION
OBSERVACION_AUDITORIA
ID_USUARIO

```
CREATE TABLE auditorias
(
  "ID_AUDITORIA" serial NOT NULL,
  "ID_EMPRESA" integer NOT NULL,
  "FECHA_CREACION" timestamp(0) without time zone NOT NULL,
  "FECHA_AUDITORIA" timestamp(0) without time zone,
  "OBSERVACION_CREACION" character varying(256),
  "OBSERVACION_AUDITORIA" character varying(256),
  "ID_USUARIO" integer NOT NULL,
  CONSTRAINT auditorias_pkey PRIMARY KEY ("ID_AUDITORIA"),
  CONSTRAINT auditorias_fk FOREIGN KEY ("ID_EMPRESA")
      REFERENCES empresas ("ID_EMPRESA") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT auditorias_fk1 FOREIGN KEY ("ID_USUARIO")
```

```
    REFERENCES usuarios ("ID_USUARIO") MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=TRUE
);
ALTER TABLE auditorias OWNER TO auditor_router;
```

Cuadro A Código de Creación Tabla Auditorias

Comandos.- En esta tabla se almacenan la descripción de los comandos con sus recomendaciones




```
CREATE TABLE comandos
(
  "ID_COMANDO" integer NOT NULL DEFAULT
nextval(('"public"."comandos_ID_COMANDO_seq"'::text)::regclass),
  "ID_TIPO_COMANDO" integer NOT NULL,
  "COMANDO" character varying(512) NOT NULL,
  "MENSAJE1" character varying(1024) NOT NULL,
  "MENSAJE2" character varying(1024) NOT NULL,
  "HABILITADO" character(1) NOT NULL,
  CONSTRAINT "PK_COMANDOS" PRIMARY KEY ("ID_COMANDO"),
```

```
  CONSTRAINT "FK1_COMANDOS" FOREIGN KEY ("ID_TIPO_COMANDO")
      REFERENCES tipos_comandos ("ID_TIPO_COMANDO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT uk1_comandos UNIQUE ("COMANDO", "HABILITADO"),
  CONSTRAINT "comandos_HABILITADO_check" CHECK ("HABILITADO" =
ANY (ARRAY['S'::bpchar, 'N'::bpchar]))
)
WITH (
  OIDS=FALSE
);
ALTER TABLE comandos OWNER TO auditor_router;
```

Cuadro B Creación Tabla Comandos

Comandos_modelos .- En esta tabla intermedia se almacena los datos comandos por modelos.




```
CREATE TABLE comandos_modelos
(
  "ID_COMANDO" integer NOT NULL,
  "ID_MODELO" integer NOT NULL,
  CONSTRAINT pk_comandos_modelos PRIMARY KEY ("ID_COMANDO",
"ID_MODELO"),
  CONSTRAINT fk1_comandos_modelos FOREIGN KEY ("ID_COMANDO")
      REFERENCES comandos ("ID_COMANDO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk2_comandos_modelos FOREIGN KEY ("ID_MODELO")
      REFERENCES modelos ("ID_MODELO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE comandos_modelos OWNER TO auditor_router;
```

Cuadro C Creación Tabla Comandos_modelos

Detalles_auditorias.- Almacena los detalles de la extracción

detalles_auditorias
ID_DETALLE_AUDITORIA
ID_AUDITORIA
ID_DISPOSITIVO_EMPRESA
CONFIGURACION

```
CREATE TABLE detalles_auditorias
(
  "ID_DETALLE_AUDITORIA" serial NOT NULL,
  "ID_AUDITORIA" integer NOT NULL,
  "ID_DISPOSITIVO_EMPRESA" integer NOT NULL,
  "CONFIGURACION" text,
  CONSTRAINT detalles_auditorias_pkey PRIMARY KEY
("ID_DETALLE_AUDITORIA"),
  CONSTRAINT detalles_auditorias_fk FOREIGN KEY ("ID_AUDITORIA")
      REFERENCES auditorias ("ID_AUDITORIA") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT detalles_auditorias_fk1 FOREIGN KEY
("ID_DISPOSITIVO_EMPRESA")
      REFERENCES dispositivos_empresas ("ID_DIPOSITIVO_EMPRESA")
MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT detalles_auditorias_idx UNIQUE ("ID_AUDITORIA",
"ID_DISPOSITIVO_EMPRESA")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE detalles_auditorias OWNER TO auditor_router;
```

Cuadro D Creación Tabla Detalles_auditorias

Detalles_auditorias_comandos.- Almacena la comparación de la extracción con los comando establecidos para la extracción.

detalles_auditorias_comandos
ID_DETALLE_AUDITORIA_COMANDO
ID_DETALLE_AUDITORIA
ID_COMANDO
CUMPLIO_COMANDO

```
CREATE TABLE detalles_auditorias_comandos
(
  "ID_DETALLE_AUDITORIA_COMANDO" integer NOT NULL DEFAULT
nextval(('"public"."detalles_auditorias_ID_DETALLE_AUDITORIA_COMANDO_s
eq"'::text)::regclass),
  "ID_DETALLE_AUDITORIA" integer NOT NULL,
  "ID_COMANDO" integer NOT NULL,
  "CUMPLIO_COMANDO" character varying(1) NOT NULL,
  CONSTRAINT "PK_DETALLES_AUDITORIAS_COMANDOS" PRIMARY
KEY ("ID_DETALLE_AUDITORIA_COMANDO"),
  CONSTRAINT "FK1_DETALLES_AUDITORIAS_COMANDOS" FOREIGN
KEY ("ID_DETALLE_AUDITORIA")
      REFERENCES detalles_auditorias ("ID_DETALLE_AUDITORIA") MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "FK2_DETALLES_AUDITORIAS_COMANDOS" FOREIGN
KEY ("ID_COMANDO")
      REFERENCES comandos ("ID_COMANDO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT uk1_detalles_auditorias_comandos UNIQUE
("ID_DETALLE_AUDITORIA", "ID_COMANDO"),
  CONSTRAINT "HHH" CHECK ("CUMPLIO_COMANDO"::text = ANY
(ARRAY['S'::character varying::text, 'N'::character varying::text]))
)
WITH (
  OIDS=FALSE
);
```

```
ALTER TABLE detalles_auditorias_comandos OWNER TO auditor_router;
```

Cuadro E Creación Tabla Detalles_auditorias_comandos

Detalles_listas_acceso.- Almacena los datos de las listas de acceso que van hacer comparadas.

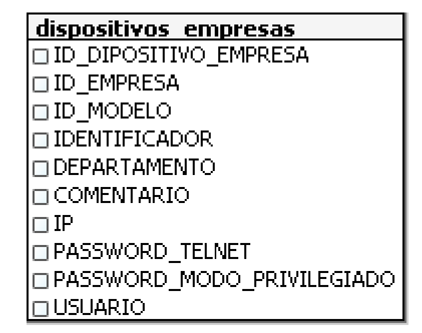
detalles_listas_acceso
ID_DETALLE_LISTA_ACCESO
ID_LISTA_ACCESO
HABILITADO
TIPO_PROTOCOLO
IP_ORIGEN
WILD_CARD_ORIGEN
IP_DESTINO
WILD_CARD_DESTINO
OPERADOR
PUERTO
NOMBRE_SERVICIO

```
CREATE TABLE detalles_listas_acceso
(
  "ID_DETALLE_LISTA_ACCESO" integer NOT NULL DEFAULT
nextval(('"public"."detalles_listas_acceso_ID_DETALLE_LISTA_ACCESO_seq"'::t
ext)::regclass),
  "ID_LISTA_ACCESO" integer NOT NULL,
  "HABILITADO" character(1) NOT NULL,
  "TIPO_PROTOCOLO" character(16) NOT NULL,
  "IP_ORIGEN" character varying(15) NOT NULL,
  "WILD_CARD_ORIGEN" character varying(15),
  "IP_DESTINO" character varying(15) NOT NULL,
  "WILD_CARD_DESTINO" character varying(15),
  "OPERADOR" character varying(16) NOT NULL,
  "PUERTO" integer,
  "NOMBRE_SERVICIO" character varying(32),
```

```
  CONSTRAINT pk_detalles_listas_acceso PRIMARY KEY ("ID_DETALLE_LISTA_ACCESO"),
  CONSTRAINT fk1_detalles_listas_acceso FOREIGN KEY ("ID_LISTA_ACCESO")
      REFERENCES listas_accesos ("ID_LISTA_ACCESO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "detalles_listas_acceso_HABILITADO_check" CHECK ("HABILITADO" = ANY (ARRAY['S'::bpchar, 'N'::bpchar]))
)
WITH (
  OIDS=FALSE
);
ALTER TABLE detalles_listas_acceso OWNER TO auditor_router;
```

Cuadro F Creación Tabla Detalles_listas_acceso

Dispositivos_empresas- Almacena los datos de los dispositivos de cada empresa con sus respectivos passwords para la extracción de los mismos.

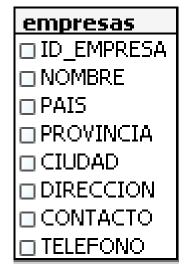



```
CREATE TABLE dispositivos_empresas
(
  "ID_DIPOSITIVO_EMPRESA" integer NOT NULL DEFAULT nextval(('"public"."dispositivos_empresas_id_dipositivo_empresa_seq"'::text)::regclass),
```

```
  "ID_EMPRESA" integer NOT NULL,
  "ID_MODELO" integer NOT NULL,
  "IDENTIFICADOR" character varying(64) NOT NULL,
  "DEPARTAMENTO" character varying(128) NOT NULL,
  "COMENTARIO" character varying(512),
  "IP" character varying(258),
  "PASSWORD_TELNET" character varying(258),
  "PASSWORD_MODO_PRIVILEGIADO" character varying(258),
  "USUARIO" character varying(64),
  CONSTRAINT dispositivos_empresas_pkey PRIMARY KEY
("ID_DIPOSITIVO_EMPRESA"),
  CONSTRAINT dispositivos_empresas_fk FOREIGN KEY ("ID_EMPRESA")
     REFERENCES empresas ("ID_EMPRESA") MATCH SIMPLE
     ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT dispositivos_empresas_fk1 FOREIGN KEY ("ID_MODELO")
     REFERENCES modelos ("ID_MODELO") MATCH SIMPLE
     ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT dispositivos_empresas_idx UNIQUE ("ID_EMPRESA",
"IDENTIFICADOR")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE dispositivos_empresas OWNER TO auditor_router;
```

Cuadro G Creación Tabla Dispositivos_empresas

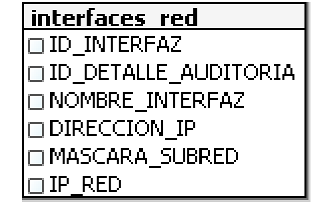
Empresas.- Almacena los datos con la identificación completa de las empresas.



```
CREATE TABLE empresas
(
  "ID_EMPRESA" integer NOT NULL DEFAULT
nextval(('"public"."empresas_id_empresa_seq"'::text)::regclass),
  "NOMBRE" character varying(64) NOT NULL,
  "PAIS" character varying(64) NOT NULL,
  "PROVINCIA" character varying(64) NOT NULL,
  "CIUDAD" character varying(64) NOT NULL,
  "DIRECCION" character varying(512) NOT NULL,
  "CONTACTO" character varying(128) NOT NULL,
  "TELEFONO" character varying(128) NOT NULL,
  CONSTRAINT empresas_pkey PRIMARY KEY ("ID_EMPRESA"),
  CONSTRAINT empresas_idx UNIQUE ("NOMBRE")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE empresas OWNER TO auditor_router;
```

Cuadro H Creación Tabla Empresas

Interfaces_red.- Almacena los datos de cada interfaz de los dispositivos que van hacer auditados.



```
CREATE TABLE interfaces_red
(
  “ID_INTERFAZ” integer NOT NULL DEFAULT
nextval((‘”public”.”interfaces_red_ID_INTERFAZ_seq”’::text)::regclass),
  “ID_DETALLE_AUDITORIA” integer NOT NULL,
  “NOMBRE_INTERFAZ” character varying(124),
  “DIRECCION_IP” character varying(15),
  “MASCARA_SUBRED” character varying(15),
  “IP_RED” character varying(15),
  CONSTRAINT interfaces_red_pkey PRIMARY KEY (“ID_INTERFAZ”),
  CONSTRAINT “interfaces_red_ID_DETALLE_AUDITORIA_fkey” FOREIGN
KEY (“ID_DETALLE_AUDITORIA”)
      REFERENCES detalles_auditorias (“ID_DETALLE_AUDITORIA”) MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT “interfaces_red_ID_DETALLE_AUDITORIA_fkey1” FOREIGN
KEY (“ID_DETALLE_AUDITORIA”)
      REFERENCES detalles_auditorias (“ID_DETALLE_AUDITORIA”) MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE interfaces_red OWNER TO auditor_router;
```

Cuadro I Creación Tabla Interfaces_red

Interfaces_red_politicas_trafico.- Almacena los datos de la buenas prácticas que se utilizaran para auditorias posteriores.

interfaces_red_politicas_trafico
ID_INTERFAZ_RED_POLITICA_TRAFICO
ID_INTERFAZ
ID_POLITICA_TRAFICO
CUMPLIO

```
CREATE TABLE interfaces_red_politicas_trafico
```

```
(
  "ID_INTERFAZ_RED_POLITICA_TRAFICO" integer NOT NULL DEFAULT
nextval(('"public"."interfaces_red_politicas_trafico_ID_INTERFAZ_RED_POLITIC
A_TRAFI"'::text)::regclass),
  "ID_INTERFAZ" integer NOT NULL,
  "ID_POLITICA_TRAFICO" integer NOT NULL,
  "CUMPLIO" character(1),
  CONSTRAINT detalles_auditorias_politicas_trafico_pkey PRIMARY KEY
("ID_INTERFAZ_RED_POLITICA_TRAFICO"),
  CONSTRAINT
"detalles_auditorias_politicas_trafico_ID_POLITICA_TRAFICO_fkey" FOREIGN
KEY ("ID_POLITICA_TRAFICO")
      REFERENCES politicas_trafico ("ID_POLITICA_TRAFICO") MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "interfaces_red_politicas_trafico_ID_INTERFAZ_fkey" FOREIGN
KEY ("ID_INTERFAZ")
      REFERENCES interfaces_red ("ID_INTERFAZ") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "detalles_auditorias_politicas_trafico_CUMPLIO_check" CHECK
("CUMPLIO" = ANY (ARRAY['S'::bpchar, 'N'::bpchar]))
)
WITH (
  OIDS=TRUE
);
ALTER TABLE interfaces_red_politicas_trafico OWNER TO auditor_router;
```

Cuadro J Creación Tabla Interfaces_red_politicas_trafico

Listas_acceso.- Almacena los datos de las listas de acceso para las auditorias del sistema.

listas_accesos
ID_LISTA_ACCESO
ID_DETALLE_AUDITORIA
NOMBRE_LISTA_ACCESO

```
CREATE TABLE listas_accesos
(
```

```
  "ID_LISTA_ACCESO" integer NOT NULL DEFAULT
nextval(('"public"."listas_accesos_ID_LISTA_ACCESO_seq"'::text)::regclass),
  "ID_DETALLE_AUDITORIA" integer NOT NULL,
  "NOMBRE_LISTA_ACCESO" character varying(128) NOT NULL,
  CONSTRAINT pk_listas_acceso PRIMARY KEY ("ID_LISTA_ACCESO"),
  CONSTRAINT fk1_listas_acceso FOREIGN KEY
("ID_DETALLE_AUDITORIA")
      REFERENCES detalles_auditorias ("ID_DETALLE_AUDITORIA") MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE listas_accesos OWNER TO auditor_router;
```

Cuadro K Creación Tabla Listas_accesos

Listas_acceso_interfaces_red.- Almacena los parámetros para ver en que sentido va la lista de acceso.

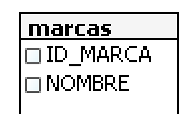
listas_accesos_interfaces_red
ID_LISTA_ACESO_INTERFAZ_RED
ID_INTERFAZ
ID_LISTA_ACCESO
SENTIDO

```
CREATE TABLE listas_accesos_interfaces_red
(
  "ID_LISTA_ACESO_INTERFAZ_RED" integer NOT NULL DEFAULT
nextval(('"public"."listas_acceso_interfaces_red_ID_LISTA_ACESO_INTERFAZ_R
ED_seq"'::text)::regclass),
  "ID_INTERFAZ" integer NOT NULL,
  "ID_LISTA_ACCESO" integer NOT NULL,
  "SENTIDO" character(1) NOT NULL,
```

```
  CONSTRAINT listas_acceso_interfaces_red_pkey PRIMARY KEY
("ID_LISTA_ACESO_INTERFAZ_RED"),
  CONSTRAINT "listas_acceso_interfaces_red_ID_INTERFAZ_fkey" FOREIGN
KEY ("ID_INTERFAZ")
      REFERENCES interfaces_red ("ID_INTERFAZ") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "listas_acceso_interfaces_red_ID_LISTA_ACCESO_fkey"
FOREIGN KEY ("ID_LISTA_ACCESO")
      REFERENCES listas_accesos ("ID_LISTA_ACCESO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "listas_acceso_interfaces_red_SENTIDO_check" CHECK
("SENTIDO" = ANY (ARRAY['I'::bpchar, 'O'::bpchar]))
)
WITH (
  OIDS=FALSE
);
ALTER TABLE listas_accesos_interfaces_red OWNER TO auditor_router;
```

Cuadro L Creación Tabla Listas_acceso_interfaces_red

Marcas.- Almacena la marca del dispositivo.




```
CREATE TABLE marcas
(
  "ID_MARCA" integer NOT NULL DEFAULT
nextval(('"public"."marcas_id_marca_seq"'::text)::regclass),
  "NOMBRE" character varying(64) NOT NULL,
  CONSTRAINT marcas_pkey PRIMARY KEY ("ID_MARCA"),
  CONSTRAINT "marcas_NOMBRE_key" UNIQUE ("NOMBRE")
)
WITH (
  OIDS=TRUE
```

```
);
ALTER TABLE marcas OWNER TO auditor_router;
```

Cuadro M Creación Tabla Marcas

Modelos.- Almacena los modelos de los dispositivos sea switch o router.

modelos
ID_MODELO
ID_MARCA
NOMBRE
ID_TIPO_DISPOSITIVO

```
CREATE TABLE modelos
(
  "ID_MODELO" integer NOT NULL DEFAULT
nextval(('"public"."modelos_id_modelo_seq"'::text)::regclass),
  "ID_MARCA" integer NOT NULL,
  "NOMBRE" character varying(64) NOT NULL,
  "ID_TIPO_DISPOSITIVO" integer NOT NULL,
  CONSTRAINT modelos_pkey PRIMARY KEY ("ID_MODELO"),
  CONSTRAINT modelos_fk FOREIGN KEY ("ID_MARCA")
      REFERENCES marcas ("ID_MARCA") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT modelos_fk1 FOREIGN KEY ("ID_TIPO_DISPOSITIVO")
      REFERENCES tipos_dispositivos ("ID_TIPO_DISPOSITIVO") MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT modelos_idx UNIQUE ("ID_MARCA", "NOMBRE",
"ID_TIPO_DISPOSITIVO")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE modelos OWNER TO auditor_router;
```

Cuadro N Creación Tabla Modelos

Políticas_trafico.- Almacena los datos de la creación de los puerto que van hacer auditados con sus recomendaciones

politicas_trafico
ID_POLITICA_TRAFICO
ID_TIPO_POLITICA_TRAFICO
TIPO_PROTOCOLO
NUMERO_PUERTO
NOMBRE_SERVICIO
DESCRIPCION_PROTOCOLO
MENSAJE_ALERTA
HABILITADO
SENTIDO

```
CREATE TABLE politicas_trafico
(
  "ID_POLITICA_TRAFICO" integer NOT NULL DEFAULT
nextval(('"public"."politicas_trafico_ID_POLITICA_TRAFICO_seq"'::text)::regclass
),
  "ID_TIPO_POLITICA_TRAFICO" integer NOT NULL,
  "TIPO_PROTOCOLO" character varying(16) NOT NULL,
  "NUMERO_PUERTO" numeric NOT NULL,
  "NOMBRE_SERVICIO" character varying(64),
  "DESCRIPCION_PROTOCOLO" character varying(512),
  "MENSAJE_ALERTA" character varying(512),
  "HABILITADO" character(1) NOT NULL,
  "SENTIDO" character(1) NOT NULL,
  CONSTRAINT "PK_politicas_trafico" PRIMARY KEY
("ID_POLITICA_TRAFICO"),
  CONSTRAINT "FK1_politicas_trafico" FOREIGN KEY
("ID_TIPO_POLITICA_TRAFICO")
      REFERENCES tipos_politicas_traficos ("ID_TIPO_POLITICA_TRAFICO")
MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "CHK1_politicas_trafico" CHECK ("HABILITADO" = ANY
(ARRAY['S'::bpchar, 'N'::bpchar])),
```

```
  CONSTRAINT "politicas_trafico_SENTIDO_check" CHECK ("SENTIDO" =
ANY (ARRAY['I'::bpchar, 'O'::bpchar]))
)
WITH (
  OIDS=FALSE
);
ALTER TABLE politicas_trafico OWNER TO auditor_router;
```

Cuadro O Creación Tabla Políticas_trafico

Políticas_trafico_modelo.- tabla intermedia para cada puerto para cada dispositivo.

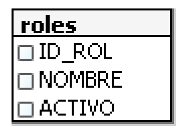
politicas_trafico_modelos
ID_POLITICA_TRAFICO
ID_MODELO

```
CREATE TABLE politicas_trafico_modelos
(
  "ID_POLITICA_TRAFICO" integer NOT NULL,
  "ID_MODELO" integer NOT NULL,
  CONSTRAINT politicas_trafico_modelos_pkey PRIMARY KEY
("ID_POLITICA_TRAFICO", "ID_MODELO"),
  CONSTRAINT "politicas_trafico_modelos_ID_MODELO_fkey" FOREIGN KEY
("ID_MODELO")
      REFERENCES modelos ("ID_MODELO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "politicas_trafico_modelos_ID_POLITICA_TRAFICO_fkey"
FOREIGN KEY ("ID_POLITICA_TRAFICO")
      REFERENCES politicas_trafico ("ID_POLITICA_TRAFICO") MATCH
SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
```

```
  OIDS=FALSE
);
ALTER TABLE politicas_trafico_modelos OWNER TO auditor_router;
```

Cuadro P Creación Tabla Políticas_trafico_modelos

Roles.- Almacena los datos la identificación de cada usuario que manejan el sistema.




```
CREATE TABLE roles
(
  "ID_ROL" serial NOT NULL,
  "NOMBRE" character varying(64) NOT NULL,
  "ACTIVO" character(1) NOT NULL,
  CONSTRAINT roles_pkey PRIMARY KEY ("ID_ROL"),
  CONSTRAINT roles_idx UNIQUE ("NOMBRE")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE roles OWNER TO auditor_router;
```

Cuadro Q Creación Tabla Roles

Tipos_comandos.- Almacena los comandos que va hacer comparados en el servidor.




```
CREATE TABLE tipos_comandos
(
  "ID_TIPO_COMANDO" integer NOT NULL DEFAULT
nextval(('"public"."tipos_comandos_ID_TIPO_COMANDO_seq"'::text)::regclass),
  "NOMBRE" character varying(64),
  CONSTRAINT "PK_TIPOS_POLITICAS" PRIMARY KEY
("ID_TIPO_COMANDO")
)
WITH (
  OIDS=FALSE
);
ALTER TABLE tipos_comandos OWNER TO auditor_router;
```

Cuadro R Creación Tabla Tipos_comandos

Tipos_dispositivos.- En la tabla almacena el nombre con el tipo de dispositivos.



```
CREATE TABLE tipos_dispositivos
(
  "ID_TIPO_DISPOSITIVO" integer NOT NULL DEFAULT
nextval(('"public"."tipos_dispositivos_id_tipo_dispositivo_seq"'::text)::regclass),
  "NOMBRE" character varying(64) NOT NULL,
  CONSTRAINT tipos_dispositivos_pkey PRIMARY KEY
("ID_TIPO_DISPOSITIVO"),
  CONSTRAINT "tipos_dispositivos_NOMBRE_key" UNIQUE ("NOMBRE")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE tipos_dispositivos OWNER TO auditor_router;
```

Cuadro S Creación Tabla Tipos_dispositivos

Tipos_politicas_traficos.- Almacena los tipos de políticas que utilizan los dispositivos en el servidor.

tipos_politicas_traficos
ID_TIPO_POLITICA_TRAFICO
NOMBRE

```
CREATE TABLE tipos_politicas_traficos
(
  "ID_TIPO_POLITICA_TRAFICO" integer NOT NULL,
  "NOMBRE" character varying(64) NOT NULL,
  CONSTRAINT "PK_TIPOS_TRAFICOS" PRIMARY KEY
("ID_TIPO_POLITICA_TRAFICO")
)
WITH (
  OIDS=FALSE
```

```
);
ALTER TABLE tipos_politicas_traficos OWNER TO auditor_router;
```

Cuadro T Creación Tabla Tipos_politicas_traficos

Usuarios.- Almacena los datos del usuario del servidor.

usuarios
ID_USUARIO
USUARIO
PASSWORD
NOMBRES
APELLIDO_PATERNO
APELLIDO_MATERNO
ACTIVO

```
CREATE TABLE usuarios
(
  "ID_USUARIO" integer NOT NULL DEFAULT
nextval(('"public"."usuarios_id_usuario_seq"'::text)::regclass),
  "USUARIO" character varying(24) NOT NULL,
  "PASSWORD" character varying(64) NOT NULL,
  "NOMBRES" character varying(80) NOT NULL,
  "APELLIDO_PATERNO" character varying(80) NOT NULL,
  "APELLIDO_MATERNO" character varying(80) NOT NULL,
  "ACTIVO" character(1) NOT NULL,
  CONSTRAINT usuarios_pkey PRIMARY KEY ("ID_USUARIO"),
  CONSTRAINT usuarios_idx UNIQUE ("USUARIO")
)
WITH (
  OIDS=TRUE
);
ALTER TABLE usuarios OWNER TO auditor_router;
```

Cuadro U Creación Tabla Usuario

Usuarios_roles.- Almacena los usuarios y quien le designo el rol usuario sea auditor o administrador.

usuarios_roles
ID_USUARIO_ROL
ID_USUARIO
ID_ROL
ID_USUARIO_ASIGNO_ROL
FECHA_INICIO_VIGENCIA
ID_USUARIO_DESASIGNO_ROL
FECHA_FIN_VIGENCIA

```
CREATE TABLE usuarios_roles
(
  "ID_USUARIO_ROL" serial NOT NULL,
  "ID_USUARIO" integer NOT NULL,
  "ID_ROL" integer NOT NULL,
  "ID_USUARIO_ASIGNO_ROL" integer NOT NULL,
  "FECHA_INICIO_VIGENCIA" timestamp(0) without time zone NOT NULL,
  "ID_USUARIO_DESASIGNO_ROL" integer,
  "FECHA_FIN_VIGENCIA" timestamp without time zone,
  CONSTRAINT usuarios_roles_pkey PRIMARY KEY ("ID_USUARIO_ROL"),
  CONSTRAINT usuarios_roles_fk FOREIGN KEY ("ID_USUARIO")
      REFERENCES usuarios ("ID_USUARIO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT usuarios_roles_fk1 FOREIGN KEY ("ID_ROL")
      REFERENCES roles ("ID_ROL") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT usuarios_roles_fk2 FOREIGN KEY
("ID_USUARIO_ASIGNO_ROL")
      REFERENCES usuarios ("ID_USUARIO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
  CONSTRAINT usuarios_roles_fk3 FOREIGN KEY
("ID_USUARIO_DESASIGNO_ROL")
      REFERENCES usuarios ("ID_USUARIO") MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=TRUE
);
ALTER TABLE usuarios_roles OWNER TO auditor_router;
```

Cuadro V Creación Tabla Usuarios_roles

1.2 Procesos Principales

1.2.1 RecuperadorConfiguracionDispositivo.java

Para la recuperación de la configuración primeramente debemos de estar conectados al dispositovivo sea este el router switch estamos utilizando variable tanto para el telnet para hacer su conexión ,sabiendo que tenemos que leer los datos utilizamos variables tanto para leer como para escribir , como debemos conectarnos atraves de un puerto, sabiendo que se ha hecho la respectiva configuración del router o switch y colocamos un usuario, password telnet y el password previlegiado utilizamos variables para esto.

La función **public** RecuperadorConfiguracionDispositivo pide los campos que hemos explicado anteriormente.

telnet.connect(ip,puerto) esta función hace canexión a traves del puerto ingresado

void escribir(String pCadena) **throws** IOException procedimiento que me permite escribir lo que se esta extraiendo del dispositivo

private String leerInterno() **throws** IOException este procedimiento nos permite leer byte por byte

private String leer() **throws** IOException, InterruptedException este procedimiento nos permite mantener todo lo leído por el procedimiento leerinterno

```
package ec.edu.ug.cisc.auditoriarouter.desktop;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStreamWriter;
import java.net.SocketException;
import org.apache.commons.net.telnet.InvalidTelnetOptionException;
import org.apache.commons.net.telnet.TelnetClient;
import org.apache.commons.net.telnet.WindowSizeOptionHandler;

public class RecuperadorConfiguracionDispositivo
{
	public final static int IGUAL=0;
	public final static int CONTIENE=1;
	public final static int INICIA=2;
	public final static int FINALIZA=3;
	private final static int SLEEP=100;

	private TelnetClient telnet;
```

```
	private OutputStreamWriter outWriter;
	private InputStream inStream;
	private String ip;
	private int puerto;
	private String usuario;
	private String passwordTelnet;
	private String passwordModoPrivilegiado;
	public RecuperadorConfiguracionDispositivo(
			String psIp,int piPuerto,String pUsuario,
			String psPasswordTelnet,String psPasswordModoPrivilegiado)
	throws SocketException, IOException, InvalidTelnetOptionException
	{
		ip=psIp;
		puerto=piPuerto;
		usuario=pUsuario;
		passwordTelnet=psPasswordTelnet;
		passwordModoPrivilegiado=psPasswordModoPrivilegiado;
		telnet=new TelnetClient();
		WindowSizeOptionHandler lWindowSizeOptionHandler=new
WindowSizeOptionHandler(Integer.MAX_VALUE,Integer.MAX_VALUE);
		telnet.addOptionHandler(lWindowSizeOptionHandler);
		telnet.connect(ip, puerto);
		outWriter=new OutputStreamWriter(telnet.getOutputStream());
		inStream=telnet.getInputStream();
	}
	public void escribir(String pCadena) throws IOException
	{
		outWriter.write(pCadena+"\r\n");
		outWriter.flush();
	}
	public String recuperar()
	throws SocketException, IOException, InterruptedException
	{
		String lsConfiguracion=null;
		boolean lbIngresoClaveTelnet=false;
		boolean lbIngresoComandoEnable=false;
		for(int i=0;i<25;i++)
		{
			Thread.sleep(SLEEP);
			String lsLectura=leer();
			if(lsLectura.contains("Press RETURN to get started!"))
				escribir("");
			else if(lsLectura.endsWith("Would you like to enter the initial
configuration dialog? [yes/no]: "))
```

```
                        escribir("no");
                else if(lsLectura.endsWith(">"))
                {
                        escribir("enable");
                        lbIngresoComandoEnable=true;
                }
                else if(lsLectura.contains("Username:"))
                {
                        escribir(usuario);
                }
                else if(lsLectura.contains("Password: "))
                {
                        if(!lbIngresoClaveTelnet)
                        {
                                escribir(this.passwordTelnet);
                                lbIngresoClaveTelnet=true;
                                lsLectura=leer();
                        }
                        else if(lbIngresoComandoEnable)
                        {
                                escribir(this.passwordModoPrivilegiado);
                                lsLectura=leer();
                        }
                }
                else if(lsLectura.endsWith("#"))
                {
                        escribir("show running-config");
                        Thread.sleep(2000);
                        lsConfiguracion=leer();

        lsConfiguracion=lsConfiguracion.substring(lsConfiguracion.indexOf("\r\n")+
2);

        lsConfiguracion=lsConfiguracion.substring(0,lsConfiguracion.indexOf("end\r\
n")+3);
                        break;
                }
                else
                        escribir("");
        }
        if(lsConfiguracion==null)
                throw new RuntimeException("No se pudo recuperar la
información del dispositivo con ip "+ip+", verifique las claves de acceso.");
        return lsConfiguracion;
```

```
	}
	private String leerInterno() throws IOException
	{
		byte []labBytes=new byte[inStream.available()];
		inStream.read(labBytes);
		return new String(labBytes);
	}
	private String leer() throws IOException, InterruptedException
	{
		Thread.sleep(SLEEP);
		StringBuilder lCadena=new StringBuilder();
		String lsString=leerInterno();
		lCadena.append(lsString.replace(" --More-- ", ""));
		while(lsString.endsWith(" --More-- "))
		{
			outWriter.write(" ");
			outWriter.flush();
			Thread.sleep(SLEEP);
			lsString=leerInterno();
			lCadena.append(lsString.substring(26).replace(" --More-- ",
""));
		}
		return lCadena.toString();
	}
	public void cerrar() throws IOException
	{
		escribir("exit");
		escribir("exit");
		escribir("exit");
		escribir("exit");
		escribir("exit");
		escribir("exit");
		telnet.disconnect();
	}
	public static void main(String arg[])
	throws SocketException, IOException, InvalidTelnetOptionException,
InterruptedException
	{
		RecuperadorConfiguracionDispositivo
lRecuperadorConfiguracionDispositivo=new RecuperadorConfiguracionDispositivo(
				"localhost", 23,"luis", "12345678", "123456");

	System.out.println(lRecuperadorConfiguracionDispositivo.recuperar());
		lRecuperadorConfiguracionDispositivo.cerrar();
```

```
        }
}
```

Cuadro W Código de RecuperadorConfiguracionDispositivo.java

1.2.2 PanelResultadoProcesamiento.java

Este proceso nos permite ver con una barra como se está extrayendo la configuración del router o switch

```
package ec.edu.ug.cisc.auditoriarouter.desktop.auditoriaRouterWizard;

import java.awt.GridBagLayout;
import java.awt.Window;
import javax.swing.JPanel;
import javax.swing.JLabel;
import java.awt.GridBagConstraints;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import javax.swing.JButton;
import ec.edu.ug.cisc.auditoriarouter.desktop.RecuperadorConfiguracionDispositivo;
import ec.edu.ug.cisc.auditoriarouter.desktop.Utils;
import java.awt.Insets;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import java.util.HashMap;
import javax.swing.JProgressBar;

public class PanelResultadoProcesamiento extends JPanel {


        /**
```

```
	 *
	 */
	private static final long serialVersionUID = -4492958296033484283L;
	private JLabel labelTitulo = null;
	private JScrollPane jspTxtResultadosProcesamiento = null;
	private JTextArea txtResultadoProcesamiento = null;
	private JPanel panelBotones = null;
	private JButton botonRegresar = null;
	private JButton botonProcesar = null;
	private JButton botonSalir = null;
	private JProgressBar jProgressBar = null;
	/**
	 * This is the default constructor
	 */
	public PanelResultadoProcesamiento() {
		super();
		initialize();
	}

	/**
	 * This method initializes this
	 *
	 * @return void
	 */
	private void initialize() {
		GridBagConstraints gridBagConstraints11 = new
GridBagConstraints();
		gridBagConstraints11.gridx = 0;
		gridBagConstraints11.fill = GridBagConstraints.BOTH;
		gridBagConstraints11.insets = new Insets(10, 50, 10, 50);
		gridBagConstraints11.gridy = 2;
		GridBagConstraints gridBagConstraints2 = new GridBagConstraints();
		gridBagConstraints2.gridx = 0;
		gridBagConstraints2.insets = new Insets(20, 0, 20, 0);
		gridBagConstraints2.gridy = 3;
		GridBagConstraints gridBagConstraints1 = new GridBagConstraints();
		gridBagConstraints1.fill = GridBagConstraints.BOTH;
		gridBagConstraints1.gridy = 1;
		gridBagConstraints1.weightx = 1.0;
		gridBagConstraints1.weighty = 1.0;
		gridBagConstraints1.insets = new Insets(20, 20, 0, 20);
		gridBagConstraints1.gridx = 0;
		GridBagConstraints gridBagConstraints = new GridBagConstraints();
		gridBagConstraints.gridx = 0;
```

```
			gridBagConstraints.insets = new Insets(20, 0, 0, 0);
			gridBagConstraints.gridy = 0;
			labelTitulo = new JLabel();
labelTitulo.setText("RESULTADOS RECUPERACION DE CONFIGURACION DE
DISPOSITIVOS");
			this.setSize(419, 228);
			this.setLayout(new GridBagLayout());
			this.add(labelTitulo, gridBagConstraints);
			this.add(getJspTxtResultadosProcesamiento(), gridBagConstraints1);
			this.add(getPanelBotones(), gridBagConstraints2);
			this.add(getJProgressBar(), gridBagConstraints11);
	}

	/**
	 * This method initializes jspTxtResultadosProcesamiento
	 *
	 * @return javax.swing.JScrollPane
	 */
	private JScrollPane getJspTxtResultadosProcesamiento() {
		if (jspTxtResultadosProcesamiento == null) {
			jspTxtResultadosProcesamiento = new JScrollPane();

	jspTxtResultadosProcesamiento.setViewportView(getTxtResultadoProcesami
ento());
		}
		return jspTxtResultadosProcesamiento;
	}

	/**
	 * This method initializes txtResultadoProcesamiento
	 *
	 * @return javax.swing.JTextArea
	 */
	private JTextArea getTxtResultadoProcesamiento() {
		if (txtResultadoProcesamiento == null) {
			txtResultadoProcesamiento = new JTextArea();
			txtResultadoProcesamiento.setEditable(false);
			txtResultadoProcesamiento.setWrapStyleWord(true);
			txtResultadoProcesamiento.setLineWrap(true);
		}
		return txtResultadoProcesamiento;
	}

	/**
```

```
	 * This method initializes panelBotones
	 *
	 * @return javax.swing.JPanel
	 */
	private JPanel getPanelBotones() {
		if (panelBotones == null) {
			GridBagConstraints gridBagConstraints5 = new GridBagConstraints();
			gridBagConstraints5.ipadx = 20;
			GridBagConstraints gridBagConstraints4 = new GridBagConstraints();
			gridBagConstraints4.insets = new Insets(0, 0, 0, 20);
			gridBagConstraints4.ipadx = 5;
			GridBagConstraints gridBagConstraints3 = new GridBagConstraints();
			gridBagConstraints3.insets = new Insets(0, 0, 0, 20);
			panelBotones = new JPanel();
			panelBotones.setLayout(new GridBagLayout());
			panelBotones.add(getBotonRegresar(), gridBagConstraints3);
			panelBotones.add(getBotonProcesar(), gridBagConstraints4);
			panelBotones.add(getBotonSalir(), gridBagConstraints5);
		}
		return panelBotones;
	}

	/**
	 * This method initializes botonRegresar
	 *
	 * @return javax.swing.JButton
	 */
	private JButton getBotonRegresar() {
		if (botonRegresar == null) {
			botonRegresar = new BotonRegresar();
		}
		return botonRegresar;
	}

	/**
	 * This method initializes botonProcesar
	 *
	 * @return javax.swing.JButton
	 */
	private JButton getBotonProcesar() {
		if (botonProcesar == null) {
```

```
botonProcesar = new JButton();
botonProcesar.setText("Procesar");
botonProcesar.addActionListener(new ActionListener()
{
        public void actionPerformed(ActionEvent e)
        {
                Thread lHilo=new Thread()
                {
                        public void run()
                        {
                                try
                                {
                                        jProgressBar.setVisible(true);

                                        jProgressBar.setIndeterminate(true);
                                        jProgressBar.setValue(0);

                                        txtResultadoProcesamiento.setText("");

txtResultadoProcesamiento.append("Inciando Procesamiento.\n");
                                        ArrayList<HashMap<String,
String>> lDetallesAuditorias=ModeloDatos.getInstance().getDetallesAuditorias();

ArrayList<HashMap<String, Object>> pConfiguraciones=new
                                        ArrayList<HashMap<String,Object>>();
                                        int
lIncremeto=90/(lDetallesAuditorias.size()*2);
                                        for(HashMap<String,
String> lDetalleAuditoria:lDetallesAuditorias)
                                        {

jProgressBar.setValue(jProgressBar.getValue()+lIncremeto);
                                        String
lsUsuario=lDetalleAuditoria.get("Usuario");
                                        String
lsIp=lDetalleAuditoria.get("Ip");
                                        String
lsPasswordTelnet=lDetalleAuditoria.get("passwordTelnet");
                                        String
lsPasswordModoPrivilegiado=lDetalleAuditoria.get("passwordModoPrivilegiado");


txtResultadoProcesamiento.append("Iniciando procesamiento del router con ip
"+lsIp+".\n");
```

```
RecuperadorConfiguracionDispositivo lrcd=null;
String
lsConfiguracion=null;
try
{
lrcd=new
RecuperadorConfiguracionDispositivo(lsIp,23,lsUsuario,lsPasswordTelnet,lsPasswor
dModoPrivilegiado);

lsConfiguracion=lrcd.recuperar();

txtResultadoProcesamiento.append("Recuperacion exitosa de la
configuracion, "+lsConfiguracion.length()+" bytes recuperados.\n");
}
finally
{
try
{

lrcd.cerrar();
}
catch
(Throwable t)
{

}
}
HashMap<String,
Object> lConfiguracion=new HashMap<String, Object>();

lConfiguracion.put("idDetalleAuditoria", lDetalleAuditoria.get("Id"));

lConfiguracion.put("configuracion", lsConfiguracion);

pConfiguraciones.add(lConfiguracion);

jProgressBar.setValue(jProgressBar.getValue()+lIncremeto);
}

txtResultadoProcesamiento.append("Enviando configuracion al servidor.\n");

ModeloDatos.getInstance().guardarConfiguraciones(pConfiguraciones);
```

```
                    txtResultadoProcesamiento.append("Configuracion almacenada con
                                                                        exito.\n");

        jProgressBar.setValue(100);
                                                        }
                                                        catch (Throwable t)
                                                        {

                        txtResultadoProcesamiento.append("Error en el procesamiento
                                                                  "+t.toString()+".\n");
                                                        }
                                                        finally
                                                        {

        botonProcesar.setEnabled(true);

        botonRegresar.setEnabled(true);

        jProgressBar.setIndeterminate(false);
                                                        }
                                        }
                                        };
                                        botonProcesar.setEnabled(false);
                                        botonRegresar.setEnabled(false);
                                        lHilo.start();
                                }
                        });
                }
                return botonProcesar;
        }

        /**
         * This method initializes botonSalir
         *
         * @return javax.swing.JButton
         */
        private JButton getBotonSalir() {
                if (botonSalir == null) {
                        botonSalir = new JButton();
                        botonSalir.setText("Salir");
                        botonSalir.addActionListener(new
java.awt.event.ActionListener() {
```

```
                                public void
actionPerformed(java.awt.event.ActionEvent e)
                                {
                                        Utils.confirmarSalir((Window)
PanelResultadoProcesamiento.this.getTopLevelAncestor());
                                }
                        });
                }
                return botonSalir;
        }

        /**
         * This method initializes jProgressBar
         *
         * @return javax.swing.JProgressBar
         */
        private JProgressBar getJProgressBar() {
                if (jProgressBar == null) {
                        jProgressBar = new JProgressBar(0,100);
                        jProgressBar.setStringPainted(true);
                        jProgressBar.setVisible(false);
                }
                return jProgressBar;
        }

}  //  @jve:decl-index=0:visual-constraint="10,10"
```

Cuadro X Código de PanelResultadoProcesamiento.java

1.2.3 AuditoriaBO.java

Este proceso nos permite realizar los objetos en los cuales vamos a trabajar.

public void crearDetalleAuditoria(DetalleAuditoria pDetalleAuditoria) sirve para realizar los detallar los cuales dispositvos se van a auditar.

public void eliminarDetallesAuditoria(Integer[] piAIdsDetallesAuditoria) sirve para eliminar detalles de auditorias.

public void crearAuditoria permite crear una nueva auditoria

public void guardarTomaMuestra(Integer[] idDetallesAuditorias,String[] configuracion) permite guardar los detalles

```
package ec.edu.ug.cisc.auditoriarouter.bo;
import java.sql.Timestamp;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Set;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import ec.edu.ug.cisc.auditoriarouter.dao.AuditoriaDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.DAO;
import ec.edu.ug.cisc.auditoriarouter.dao.DetalleAuditoriaComandoDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.DetalleAuditoriaDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.InterfazRedPoliticaTraficoDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.DetalleListaAccesoDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.DispositivoEmpresaDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.InterfazRedDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.ListaAccesoDAO;
import ec.edu.ug.cisc.auditoriarouter.dao.ListaAccesoInterfazRedDAO;
import ec.edu.ug.cisc.auditoriarouter.entity.Auditoria;
import ec.edu.ug.cisc.auditoriarouter.entity.Comando;
import ec.edu.ug.cisc.auditoriarouter.entity.DetalleAuditoria;
import ec.edu.ug.cisc.auditoriarouter.entity.DetalleAuditoriaComando;
import ec.edu.ug.cisc.auditoriarouter.entity.DetalleListaAcceso;
import ec.edu.ug.cisc.auditoriarouter.entity.DispositivoEmpresa;
```

```
import ec.edu.ug.cisc.auditoriarouter.entity.InterfazRed;
import ec.edu.ug.cisc.auditoriarouter.entity.InterfazRedPoliticaTrafico;
import ec.edu.ug.cisc.auditoriarouter.entity.ListaAcceso;
import ec.edu.ug.cisc.auditoriarouter.entity.ListaAccesoInterfazRed;
import ec.edu.ug.cisc.auditoriarouter.entity.Modelo;
import ec.edu.ug.cisc.auditoriarouter.entity.PoliticaTrafico;
import ec.edu.ug.cisc.auditoriarouter.excepciones.ExcepcionLogicaNegocio;
import ec.edu.ug.cisc.auditoriarouter.excepciones.UtilExcepciones;

public class AuditoriaBO
{
	public final static String EQ="EQ";
	public final static String NEQ="NEQ";
	public final static String GT="GT";
	public final static String LT="LT";

	public void crearAuditoria(Auditoria pAuditoria)
	{
		try
		{
			DAO.beginTransaction();
	pAuditoria.setFechaCreacion(new Timestamp(System.currentTimeMillis()));
			AuditoriaDAO lAuditoriaDAO=new AuditoriaDAO();
			lAuditoriaDAO.persist(pAuditoria);
			DAO.commitTransaction();
		}
		catch (Exception e)
		{
			DAO.rollbackTransaction();
			throw UtilExcepciones.transformar(e);
		}
	}

	public void crearDetalleAuditoria(DetalleAuditoria pDetalleAuditoria)
	{
		try
		{
			DAO.beginTransaction();
				DetalleAuditoriaDAO lDetalleAuditoriaDAO=new
						DetalleAuditoriaDAO();
						DetalleAuditoria
 lDAAux=lDetalleAuditoriaDAO.findDetalleAuditoriaByIdAuditoriaIdDispositivo(

	pDetalleAuditoria.getAuditoria().getIdAuditoria(),
```

```
				pDetalleAuditoria.getDispositivoEmpresa().getIdDipositivoEmpresa());

			if(lDAAux!=null)
				throw new ExcepcionLogicaNegocio("Actualmente ya existe registrado el dispositivo con identificador "+lDAAux.getDispositivoEmpresa().getIdentificador()+" para la auditoria "+pDetalleAuditoria.getAuditoria().getIdAuditoria(),
						"Violacion de clave unica");

				DispositivoEmpresaDAO lDispositivoEmpresaDAO=new
								DispositivoEmpresaDAO();
				DispositivoEmpresa
lDispositivoEmpresa=lDispositivoEmpresaDAO.findById(pDetalleAuditoria.getDisp
		ositivoEmpre	sa().getIdDipositivoEmpresa());
			if(lDispositivoEmpresa.getModelo().getComandos().size()==0)
				throw new ExcepcionLogicaNegocio(
					"El dispositivo "+lDispositivoEmpresa.getIdentificador()+" corresponde a un modelo sin comandos a auditar.\nModelo "+lDispositivoEmpresa.getModelo().getNombre()+" sin comandos a auditar.",
					"Modelo sin comandos a auditar.");

			lDetalleAuditoriaDAO.persist(pDetalleAuditoria);
			DAO.commitTransaction();
		}
		catch (Exception e)
		{
			DAO.rollbackTransaction();
			throw UtilExcepciones.transformar(e);
		}
	}
	public void eliminarDetallesAuditoria(Integer[] piAIdsDetallesAuditoria)
	{
		try
		{
			DAO.beginTransaction();
				DetalleAuditoriaDAO lDetalleAuditoriaDAO=new
								DetalleAuditoriaDAO();
			for(Integer idDetalleAuditoria:piAIdsDetallesAuditoria)
			{
				DetalleAuditoria lDetalleAuditoria=lDetalleAuditoriaDAO.
							findById(idDetalleAuditoria);
				lDetalleAuditoriaDAO.remove(lDetalleAuditoria);
```

```
                        }
                        DAO.commitTransaction();
                }
                catch (Exception e)
                {
                        DAO.rollbackTransaction();
                        throw UtilExcepciones.transformar(e);
                }
        }
        public void guardarTomaMuestra(Integer[] idDetallesAuditorias,String[]
configuracion)
        {
                try
                {
                        DAO.beginTransaction();
                        Timestamp lHoy=new
Timestamp(System.currentTimeMillis());
                                      DetalleAuditoriaDAO lDetalleAuditoriaDAO=new
                                                                DetalleAuditoriaDAO();
                        for(int i=0;i<idDetallesAuditorias.length;i++)
                        {
                                                                        DetalleAuditoria
          lDetalleAuditoria=lDetalleAuditoriaDAO.findById(idDetallesAuditorias[i]);

        lDetalleAuditoria.getAuditoria().setFechaAuditoria(lHoy);
                                lDetalleAuditoria.setConfiguracion(configuracion[i]);
                                lDetalleAuditoriaDAO.merge(lDetalleAuditoria);
                        }
                        DAO.commitTransaction();
                }
                catch (Exception e)
                {
                        DAO.rollbackTransaction();
                        throw UtilExcepciones.transformar(e);
                }
        }

        @SuppressWarnings("unchecked")
        public void auditar(Integer pIdAuditoria)
        {
                try
                {
                        DAO.beginTransaction();
```

```
DetalleAuditoriaComandoDAO
lDetalleAuditoriaComandoDAO=new DetalleAuditoriaComandoDAO();
InterfazRedPoliticaTraficoDAO
lInterfazRedPoliticaTraficoDAO=new InterfazRedPoliticaTraficoDAO();
AuditoriaDAO lAuditoriaDAO=new AuditoriaDAO();
ListaAccesoDAO lListaAccesoDAO=new ListaAccesoDAO();
DetalleListaAccesoDAO lDetalleListaAccesoDAO=new
DetalleListaAccesoDAO();
```

```
InterfazRedDAO lInterfazRedDAO=new InterfazRedDAO();
ListaAccesoInterfazRedDAO
lListaAccesoInterfazRedDAO=new ListaAccesoInterfazRedDAO();

Auditoria lAuditoria=lAuditoriaDAO.findById(pIdAuditoria);
for(DetalleAuditoria
lDetalleAuditoria:lAuditoria.getDetallesAuditoria())
{
    if(lDetalleAuditoria.getConfiguracion()==null)
        throw new ExcepcionLogicaNegocio("No ha
sido tomada la muestra de los datos de los dispositivos para la auditoria
"+pIdAuditoria+".");
String lStrConfiguracion=lDetalleAuditoria.getConfiguracion().toUpperCase();
Modelo lModelo=lDetalleAuditoria.getDispositivoEmpresa().getModelo();
    Set<Comando> lComandos=lModelo.getComandos();
    if(lComandos.size()==0)
    {
        throw new ExcepcionLogicaNegocio(
"El dispositivo "+lDetalleAuditoria.getDispositivoEmpresa().getIdentificador()+"
corresponde a un modelo sin comandos a auditar.\nModelo
"+lModelo.getNombre()+" sin comandos a auditar.",
"Modelo sin comandos a auditar.");
    }
        lDetalleAuditoriaComandoDAO.remove(new
ArrayList(lDetalleAuditoria.getDetalleAuditoriaComandos()));
    lDetalleAuditoriaComandoDAO.flush();

    for(InterfazRed
lInterfazRed:lDetalleAuditoria.getInterfacesRed())
    {
        lListaAccesoInterfazRedDAO.remove(new
ArrayList(lInterfazRed.getListasAccesoInterfacesRed()));
        lListaAccesoInterfazRedDAO.flush();

        lInterfazRedPoliticaTraficoDAO.remove(new
ArrayList(lInterfazRed.getInterfazRedPoliticasTrafico()));
```

```
                    lInterfazRedPoliticaTraficoDAO.flush();
                }

    lInterfazRedDAO.remove(new ArrayList(lDetalleAuditoria.getInterfacesRed()));
                lInterfazRedDAO.flush();

                for(Comando lComando:lComandos)
                {
            String lStrComando=lComando.getComando().toUpperCase();
                                                        boolean
    lComandoHabilitado=!lStrConfiguracion.contains("NO "+lStrComando) &&
                        lStrConfiguracion.contains(lStrComando);

        DetalleAuditoriaComando lDetalleAuditoriaComando=new
                DetalleAuditoriaComando();

    lDetalleAuditoriaComando.setComando(lComando);

    lDetalleAuditoriaComando.setDetalleAuditoria(lDetalleAuditoria);

    lDetalleAuditoriaComando.setCumplioComando(lComando.estaHabilitado()=
                                        =lComandoHabilitado);

lDetalleAuditoria.getDetalleAuditoriaComandos().add(lDetalleAuditoriaComando);


            lDetalleAuditoriaComandoDAO.persist(lDetalleAuditoriaComando);
                }

lStrConfiguracion=lStrConfiguracion.replace(" ANY", " 0.0.0.0 255.255.255.255");
                lStrConfiguracion=lStrConfiguracion.replaceAll(" +","
");

                String lsPatronACLExtendida="access-list ((1[0-9][0-
    9])|(2[0-6][0-9][0-9])|(\\p{Alnum}+)) ((permit)|(deny)) ((udp)|(tcp)|(ip)|(icmp))
    (\\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3} \\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3})
    (\\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3} \\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3})
                            ((eq)|(neq)|(gt)|(lt)) \\p{Print}+";

                        Pattern
lPatronIdentificarACLExtendidas=Pattern.compile(lsPatronACLExtendida,Pattern.C
                ASE_INSENSITIVE);
```

```
Matcher
lMatcherIdentificarACLExtendidas=lPatronIdentificarACLExtendidas.matcher(lStrC
onfiguracion);

HashMap<String,ListaAcceso> lListasAcceso=new
HashMap<String,ListaAcceso>();

while(lMatcherIdentificarACLExtendidas.find())
{
String[] lCamposACL=lMatcherIdentificarACLExtendidas.group().split(" ");
String lsNombreListaAcceso=lCamposACL[1];
String lsHabilitado=lCamposACL[2].equalsIgnoreCase("permit")?"S":"N";
String lsTipoProtocolo=lCamposACL[3];
String lsIpOrigen=lCamposACL[4];
String lsWildCardOrigen=lCamposACL[5];
String lsIpDestino=lCamposACL[6];
String lsWildCardDestino=lCamposACL[7];
String lsOperador=lCamposACL[8];
String
lsPuertoNombreServicio=lCamposACL[9];
ListaAcceso lListaAcceso=lListasAcceso.get(lsNombreListaAcceso);
if(lListaAcceso==null)
{
lListaAcceso=new ListaAcceso();

lListaAcceso.setNombreListaAcceso(lsNombreListaAcceso);

lListaAcceso.setDetalleAuditoria(lDetalleAuditoria);

lDetalleAuditoria.getListasAccesos().add(lListaAcceso);
lListaAccesoDAO.persist(lListaAcceso);
lListasAcceso.put(lsNombreListaAcceso, lListaAcceso);
}

DetalleListaAcceso lDetalleListaAcceso=new DetalleListaAcceso();
lDetalleListaAcceso.setHabilitado(lsHabilitado);

lDetalleListaAcceso.setTipoProtocolo(lsTipoProtocolo);
lDetalleListaAcceso.setIpOrigen(lsIpOrigen);
lDetalleListaAcceso.setWildCardOrigen(lsWildCardOrigen);
lDetalleListaAcceso.setIpDestino(lsIpDestino);
lDetalleListaAcceso.setWildCardDestino(lsWildCardDestino);
lDetalleListaAcceso.setOperador(lsOperador);
```

```
try
{
    lDetalleListaAcceso.setPuerto(new Integer(lsPuertoNombreServicio));
}
catch (NumberFormatException e)
{
    lDetalleListaAcceso.setNombreServicio(lsPuertoNombreServicio);
}

lDetalleListaAcceso.setListaAcceso(lListaAcceso);

lListaAcceso.getDetallesListaAcceso().add(lDetalleListaAcceso);

lDetalleListaAccesoDAO.persist(lDetalleListaAcceso);
}
lDetalleListaAccesoDAO.flush();

String lsPatronIdentificarInterfaces="interface \\p{Print}+[^!]+!";
Pattern lPatronIdentificarInterfaces=Pattern.compile(lsPatronIdentificarInterfaces,Pattern.DOTALL|Pattern.CASE_INSENSITIVE);
Matcher lMatcherIdentificarInterfaces=lPatronIdentificarInterfaces.matcher(lDetalleAuditoria.getConfiguracion());
while(lMatcherIdentificarInterfaces.find())
{
String lsTextoDatosInterfaz=lMatcherIdentificarInterfaces.group();
String lNombreInterfaz=lsTextoDatosInterfaz.split(" ")[1].replace("\n", "");

String lsPatronIdentificarLineaAddress="ip address \\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3} \\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3}";
Pattern lPatronIdentificarLineaAddress=Pattern.compile(lsPatronIdentificarLineaAddress,Pattern.CASE_INSENSITIVE);
Matcher lMatcherIdentificarLineaAddress=lPatronIdentificarLineaAddress.matcher(lsTextoDatosInterfaz);
```

```
String lsDireccionIP=null;
String lsMascaraSubred=null;
String lsIPRed=null;

if(lMatcherIdentificarLineaAddress.find())
{
String lDatosInterfaz[]=lMatcherIdentificarLineaAddress.group().split(" ");
lsDireccionIP=lDatosInterfaz[2];
lsMascaraSubred=lDatosInterfaz[3];

String[] lsOctetosDireccionIP=lsDireccionIP.split("\\.");
String[] lsOctetosMascaraSubred=lsMascaraSubred.split("\\.");

lsIPRed="";
for(int i=0;i<4;i++)
{
int lOctetoDireccionIp=Integer.parseInt(lsOctetosDireccionIP[i]);
int lOctetoMascaraSubred=Integer.parseInt(lsOctetosMascaraSubred[i]);
int lOctetoDireccionRed=lOctetoDireccionIp&lOctetoMascaraSubred;

if(i!=3)
lsIPRed=lsIPRed+lOctetoDireccionRed+".";
else
lsIPRed=lsIPRed+lOctetoDireccionRed;
}
}

InterfazRed lInterfazRed=new InterfazRed();
lInterfazRed.setDireccionIp(lsDireccionIP);
lInterfazRed.setIpRed(lsIPRed);

lInterfazRed.setMascaraSubred(lsMascaraSubred);

lInterfazRed.setNombreInterfaz(lNombreInterfaz);

lInterfazRed.setDetalleAuditoria(lDetalleAuditoria);

lDetalleAuditoria.getInterfacesRed().add(lInterfazRed);

lInterfazRedDAO.persist(lInterfazRed);
lInterfazRedDAO.flush();

String lsPatronIdentificarAccesGroup="ip access-group \\p{Print}+ ((out)|(in))";
```

```
Pattern
lPatronIdentificarAccesGroup=Pattern.compile(lsPatronIdentificarAccesGroup,Pattern.CASE_INSENSITIVE);
Matcher
lMatcherIdentificarAccesGroup=lPatronIdentificarAccesGroup.matcher(lsTextoDatosInterfaz);
while(lMatcherIdentificarAccesGroup.find())
{
String lsDatosAccesGroup=lMatcherIdentificarAccesGroup.group();
String lsNombreAccesList=lsDatosAccesGroup.split(" ")[2];
String lsSentido=lsDatosAccesGroup.split(" ")[3].toLowerCase().equals("out")?"O":"I";

ListaAcceso lListaAcceso=lListasAcceso.get(lsNombreAccesList);
if(lListaAcceso==null)
continue;

ListaAccesoInterfazRed lListaAccesoInterfazRed=new ListaAccesoInterfazRed();
lInterfazRed.getListasAccesoInterfacesRed().add(lListaAccesoInterfazRed);

lListaAccesoInterfazRed.setInterfazRed(lInterfazRed);

lListaAccesoInterfazRed.setSentido(lsSentido);

lListaAccesoInterfazRed.setListaAcceso(lListaAcceso);

lListaAccesoInterfazRedDAO.persist(lListaAccesoInterfazRed);
}

Set<PoliticaTrafico>
lPoliticasTrafico=lModelo.getPoliticasTrafico();
if(lPoliticasTrafico.size()==0)
{
throw new ExcepcionLogicaNegocio(
"El dispositivo "+lDetalleAuditoria.getDispositivoEmpresa().getIdentificador()+" corresponde a un modelo sin politicas de trafico a auditar.\nModelo "+lModelo.getNombre()+" sin politicas de trafico configuradas.","Modelo sin politicas de trafico a auditar.");
}
for(PoliticaTrafico
lPoliticaTrafico:lPoliticasTrafico)
{
ListaAcceso
lListaAcceso=lListaAccesoDAO.findListaAccesoByIdInterfazAndSentido(
```

```
lInterfazRed.getIdInterfaz(), lPoliticaTrafico.getSentido());

                                        boolean lCumplio=false;
                                        if(lListaAcceso!=null)
                                        {
                                                String lWildCard=null;
                    for(String lOctetoIpRed:lInterfazRed.getMascaraSubred().split("\\."))
                                                {
                                    int liNumero=Integer.parseInt(lOctetoIpRed);
                                                        int liBase=255;
                    int lOctetoWildCard=(liNumero|liBase) & (~(liNumero&liBase));

                                                        if(lWildCard==null)

                                                lWildCard=lOctetoWildCard+"";
                                                        else

                                    lWildCard=lWildCard+"."+lOctetoWildCard;
                                                }

                                                            List<DetalleListaAcceso>
                        lDetallesListaAcceso=new ArrayList<DetalleListaAcceso>();

                                                String lsIpAll="0.0.0.0";
                                                                        String
                                                lsWildCardAll="255.255.255.255";

                                                String lsIpOrigen=lsIpAll;
                                                String
lsWildCardOrigen=lsWildCardAll;
                                                String lsIpDestino=lsIpAll;
                                        String lsWildCardDestino=lsWildCardAll;

                                                if(lPoliticaTrafico.isSentidoIn())
                                                {
                                                lsIpOrigen=lInterfazRed.getIpRed();


        lsWildCardOrigen=lWildCard;
                                                }
                                                else
if(lPoliticaTrafico.isSentidoOut())
                                                {
                                                lsIpDestino=lInterfazRed.getIpRed();
```

```
lsWildCardDestino=lWildCard;
                                }
                                else
                                    throw new
IllegalStateException("Sentido de la politica de trafico
"+lPoliticaTrafico.getIdPoliticaTrafico()+" incorrecta.");


lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndPuerto(
                                lListaAcceso.getIdListaAcceso(),
        lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                lsIpOrigen, lsWildCardOrigen,
                                lsIpDestino, lsWildCardDestino,
                        EQ, lPoliticaTrafico.getNumeroPuerto()));




lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndPuerto(
                                lListaAcceso.getIdListaAcceso(),
        lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                        lsIpAll,
lsWildCardAll,
                                        lsIpAll,
lsWildCardAll,
    EQ, lPoliticaTrafico.getNumeroPuerto()));

lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndLTPuerto(

lListaAcceso.getIdListaAcceso(),

lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                lsIpOrigen, lsWildCardOrigen,
                                lsIpDestino, lsWildCardDestino,
                        GT, lPoliticaTrafico.getNumeroPuerto()));
```

```
        lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndLTPuerto(
                                                  lListaAcceso.getIdListaAcceso(),
                 lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                                                 lsIpAll,
lsWildCardAll,
                                                                 lsIpAll,
lsWildCardAll,
                                        GT, lPoliticaTrafico.getNumeroPuerto()));
```

```
        lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndGTPuerto(
                                                  lListaAcceso.getIdListaAcceso(),
                 lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                                    lsIpOrigen, lsWildCardOrigen,
                                                   lsIpDestino, lsWildCardDestino,
                                        LT, lPoliticaTrafico.getNumeroPuerto()));


        lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndGTPuerto(
                                                  lListaAcceso.getIdListaAcceso(),
                 lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                                                 lsIpAll,
lsWildCardAll,
                                                                 lsIpAll,
lsWildCardAll,
                                        LT, lPoliticaTrafico.getNumeroPuerto()));

                 if(lPoliticaTrafico.getNombreServicio()!=null &&
               !lPoliticaTrafico.getNombreServicio().trim().equals(""))
                                                 {

        lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAcces
oListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrig
enAndIpDestinoAndWildCardDestinoAndOperadorAndNombreServicio(
                                                  lListaAcceso.getIdListaAcceso(),
                 lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
                                                    lsIpOrigen, lsWildCardOrigen,
```

```
lsIpDestino, lsWildCardDestino,
EQ, lPoliticaTrafico.getNombreServicio()));


lDetallesListaAcceso.addAll(lDetalleListaAccesoDAO.findDetalleListaAccesoListIdListaAccesoAndHabilitadoAndTipoProtocoloAndIpOrigenAndWildCardOrigenAndIpDestinoAndWildCardDestinoAndOperadorAndNombreServicio(
lListaAcceso.getIdListaAcceso(),
lPoliticaTrafico.getHabilitado(), lPoliticaTrafico.getTipoProtocolo(),
lsIpAll, lsWildCardAll,
lsIpAll, lsWildCardAll,
EQ, lPoliticaTrafico.getNombreServicio()));
}

lCumplio=lDetallesListaAcceso.size()>0;
}
InterfazRedPoliticaTrafico lInterfazRedPoliticaTrafico=new
InterfazRedPoliticaTrafico();
lInterfazRedPoliticaTrafico.setInterfazRed(lInterfazRed);
lInterfazRedPoliticaTrafico.setPoliticaTrafico(lPoliticaTrafico);
lInterfazRedPoliticaTrafico.setCumplio(lCumplio);

lInterfazRedPoliticaTraficoDAO.persist(lInterfazRedPoliticaTrafico);
}
}

}
DAO.commitTransaction();
}
catch (Exception e)
{
DAO.rollbackTransaction();
throw UtilExcepciones.transformar(e);
}
}

public boolean actualizarContenido(Integer pIdAuditoria,String pIdentificador,String
pConfiguracion)
{ try
{
DAO.beginTransaction();

DetalleAuditoriaDAO lDetalleAuditoriaDAO=new DetalleAuditoriaDAO();
```

```
DetalleAuditoria
lDetalleAuditoria=lDetalleAuditoriaDAO.findDetalleAuditoriaByIdAuditoriaAndIde
ntificador(pIdAuditoria, pIdentificador);
			if(lDetalleAuditoria!=null)
			{
				lDetalleAuditoria.setConfiguracion(pConfiguracion);
				lDetalleAuditoriaDAO.merge(lDetalleAuditoria);
				DAO.commitTransaction();
			}
			return lDetalleAuditoria!=null;
		}
		catch (Exception e)
		{
			DAO.rollbackTransaction();
			throw UtilExcepciones.transformar(e);
		}
	}

	public static void main(String arg[])
	{
		//byte lOcteto=(byte) 0xff;
		int liNumero=48;
		int liBase=255;
		int liOcteto=(liNumero|liBase) & (~(liNumero&liBase));
		System.out.println(liOcteto);

		//System.out.println(new Integer() Integer.toHexString(255));
	}
}
```

Cuadro Y Código de AuditoriaBO.java

1.5.4 CargadorArchivos.java

Este proceso nos permite extraer los datos poniendolo en un archivo .log en la carpeta.

public void contextDestroyed(ServletContextEvent pServletContextEvent) este procedimento sirve para terminar el proceso.

public void contextInitialized(ServletContextEvent pServletContextEvent) sirve para inicar con los eventos.

```
package ec.edu.ug.cisc.auditoriarouter.web;

import java.io.File;
import java.io.FileFilter;
import java.io.FileInputStream;

import javax.servlet.ServletContextEvent;
import javax.servlet.ServletContextListener;

import ec.edu.ug.cisc.auditoriarouter.bo.AuditoriaBO;
import ec.edu.ug.cisc.auditoriarouter.dao.DAO;

public class CargadorArchivos implements ServletContextListener
{
	private File carpeta;
	private boolean terminar=false;
	private Thread hilo=new Thread()
	{
		@Override
		public void run()
		{
			while(true)
			{
				if(terminar)
					break;
				File [] lArchivos=carpeta.listFiles(new FileFilter()
				{
					public boolean accept(File pathname)
					{
				   return pathname.getName().matches("\\d+_.+\\.txt");
					}
```

```
                                        });
                                        try
                                        {
                                                sleep(2000);
                                        }
                                        catch (InterruptedException e1)
                                        {
                                                e1.printStackTrace();
                                        }
                                        for(File lArchivo:lArchivos)
                                        {
                                                if(terminar)
                                                        break;
                                                try
                                                {
                                                        String
lsNombreArchivo=lArchivo.getName();
                                          int liIndiceSubguion=lsNombreArchivo.indexOf("_");
                                           int liIndicePunto=lsNombreArchivo.lastIndexOf(".");
 Integer lIdAuditoria=new Integer(lsNombreArchivo.substring(0, liIndiceSubguion));
String lIdentificador=lsNombreArchivo.substring(liIndiceSubguion+1,liIndicePunto);

                                            FileInputStream lio=new FileInputStream(lArchivo);
                                                        byte[] lLongitud=new
byte[lio.available()];
                                                        lio.read(lLongitud);
                                                        lio.close();
                                                AuditoriaBO lAuditoriaBO=new AuditoriaBO();
                                        if(lAuditoriaBO.actualizarContenido(lIdAuditoria,
lIdentificador, new String(lLongitud)))
                                                        {
                File lArchivoRenombrar=new File(carpeta,lArchivo.getName()+".prc");
                                                                lArchivoRenombrar.delete();
                                                        lArchivo.renameTo(lArchivoRenombrar);
                                                        }
                                                }
                                                catch (Exception e)
                                                {
                                                        e.printStackTrace();
                                                }
                                        }
                                }
                        }
                };
```

```
	public void contextDestroyed(ServletContextEvent pServletContextEvent)
	{
		terminar=true;
	}

	public void contextInitialized(ServletContextEvent pServletContextEvent)
	{
		Object lRuta=pServletContextEvent.getServletContext().getInitParameter("rutaSFTP");
		DAO.beginTransaction();
		if(lRuta!=null)
		{
			carpeta=new File(lRuta.toString());
			terminar=false;
			hilo.start();
		}
	}
}
```

Cuadro Z CargadorArchivos.java

CAPÍTULO 2

2.1 Introduccion

A continuación se detallará el contenido de cada una de las opciones que el sistema de auditoría de seguridades de router y switch cisco vía web que posee. Cabe mencionar la normalización de procesos. Estos procesos estarán en cada una de las pantallas del sistema por lo que procederemos a explicarlos para una mejor utilización de los mismos.

2.2 La Auditoria

Nos permite constatar si acaso lo que se está teniendo en funcionamiento está cumpliendo con normas y expectativas de la empresa arrojando los resultados deseados.

2.3 La Auditoria Sistematizada

El mundo se va adaptando a grandes tecnología y es por eso que este sistemas arroja los reportes de cómo están los router y switch cisco vía web con sus políticas de tráfico y sus comandos arrojando una exactitud con un margen de error del 0.01 %.

2.4 Hardware y Software Requeridos

Este sistema necesita tener por lo mínimo 1 Gb de memoria RAM, sistema operativo Windows Xp, Vista, Server. Disco duro de 80 Gb como mínimo

2.5 Contenido del Manual

Se ha dividido en 6 capítulos para un completo conocimiento del producto se precisa la lectura intima del manual, su segmentación será de utilidad para la realización de posteriores consultas de forma rápida.

2.6 Primera Instalacion

Para realizar la primera instalación debemos de tener todos los requerimientos de hardware y software tenemos que levantar los servicios tanto del tomcat y ver si la base de datos está bien instalada con los eclipse abrimos el programa y lo generamos en paquetes .jar instalados tanto al servidor como también a los que van a ser auditores

2.7 Limite de Usuarios

El sistema es ilimitado

2.8 Acceso

Contiene los accesos al sistema en este caso son dos Administrador y Auditor

2.8.1 Sistema de Acceso

En esta parte vale recalcar que hay 2 usuarios el uno es el administrador y el otro el auditor

2.8.2 Niveles de Acceso

PANTALLA DE INICIO

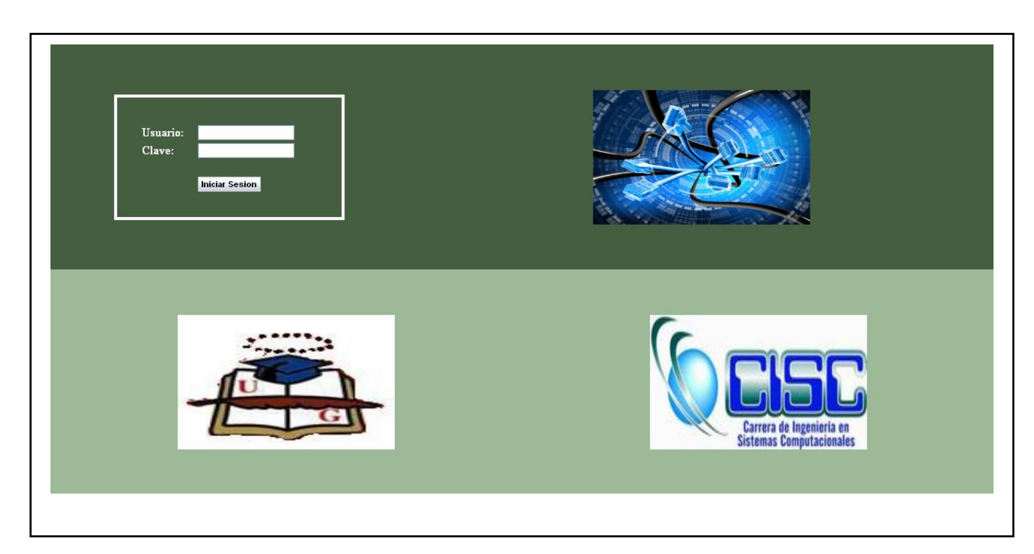


Figura A Pantalla Inicio

Se ingresa el usuario y contraseña dadas por el administrador del sistema se aplasta el botón Aceptar

PANTALLA DEL ADMINISTRADOR



Figura B Pantalla del Administrador

El administrador es que controla todas las opciones del sistema caso crear usuarios, empresas, comandos, leyes de tráfico, router o switch, e inclusive puede realizar la auditoria extrayendo un archivo vía TFTP estando desde el servidor

PANTALLA DEL AUDITOR

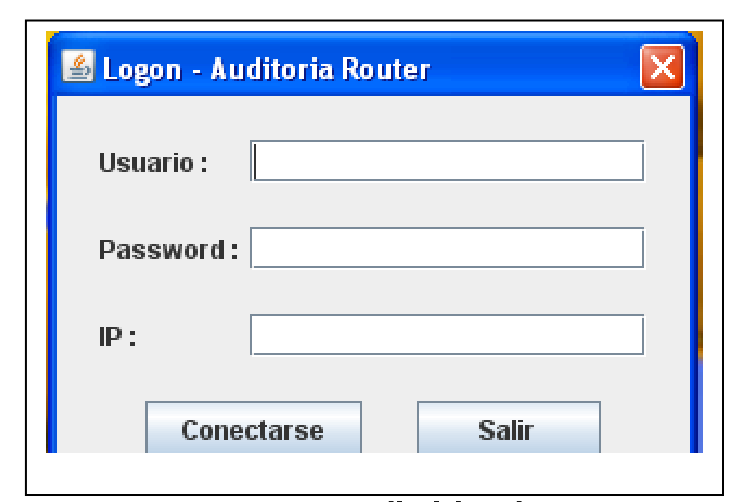


Figura C   Pantalla del Auditor

El auditor ingresa con su clave, usuario y la dirección IP del servidor de la empresa que va a realizar la auditoria

2.9 Procesos

2.9.1 Conectarse a Router y/o Switch

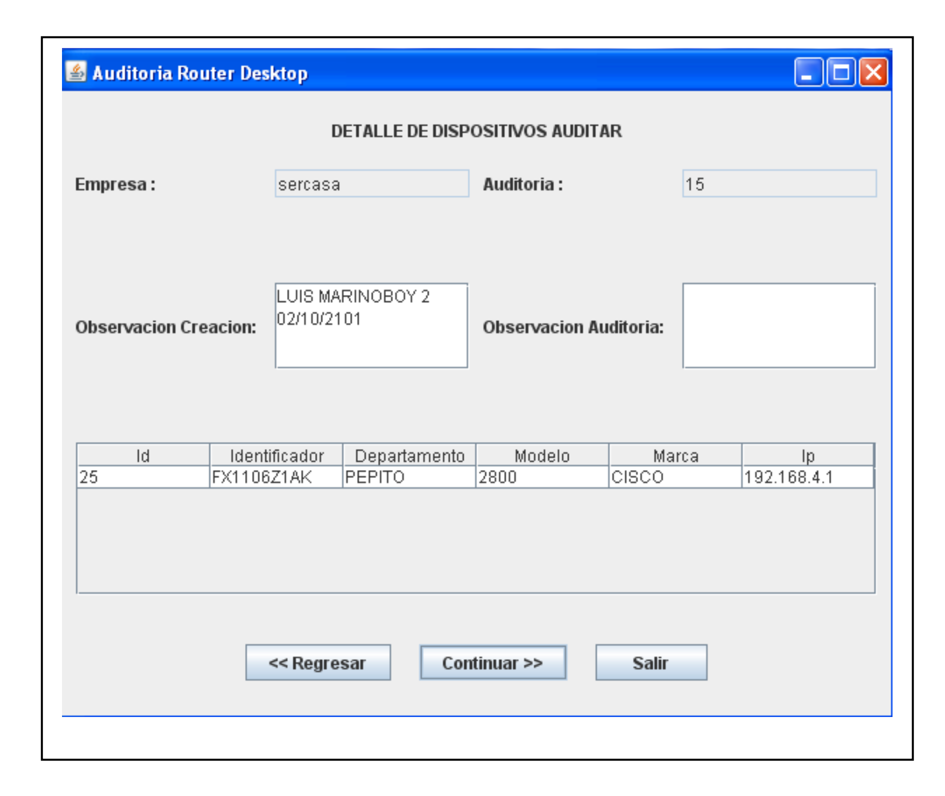


Figura D1 Pantalla Conectarse a Router y/o Switch

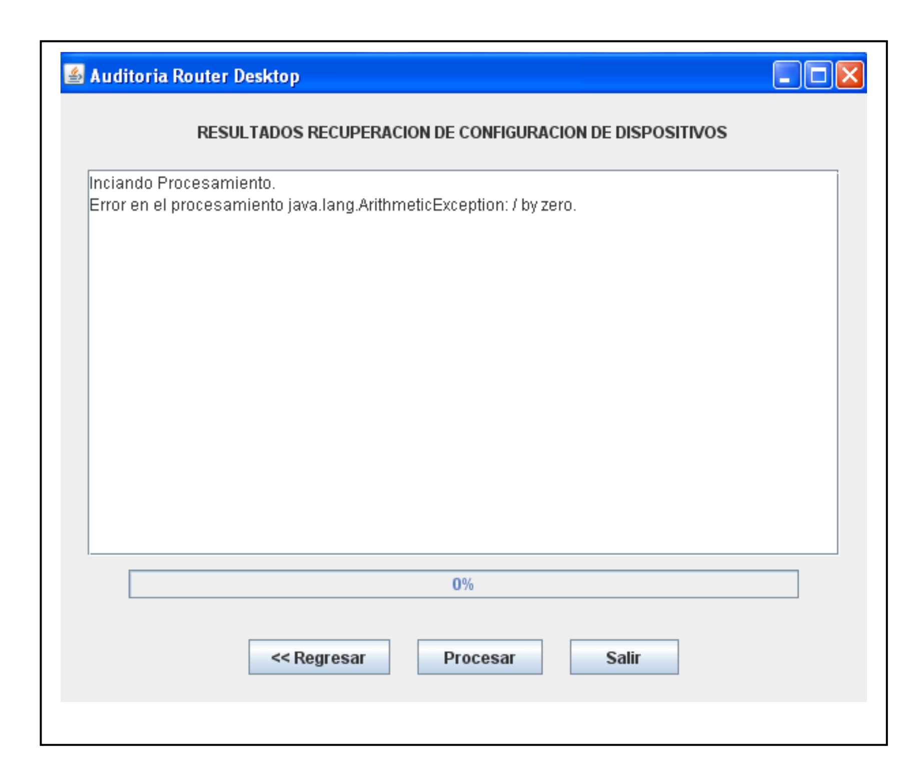


Figura D2 Pantalla Conectarse a Router y/o Switch

Se conecta al router o switch siempre y cuando lo realice el auditor asignado previamente el administrador al crear la auditoria de la dirección ip por la cual puede accesar para realizar la conexión con sus router o switch también les asigna las claves del telnet y del dispositivo.

2.9.2 Extracción De Datos (telnet o TFTP)

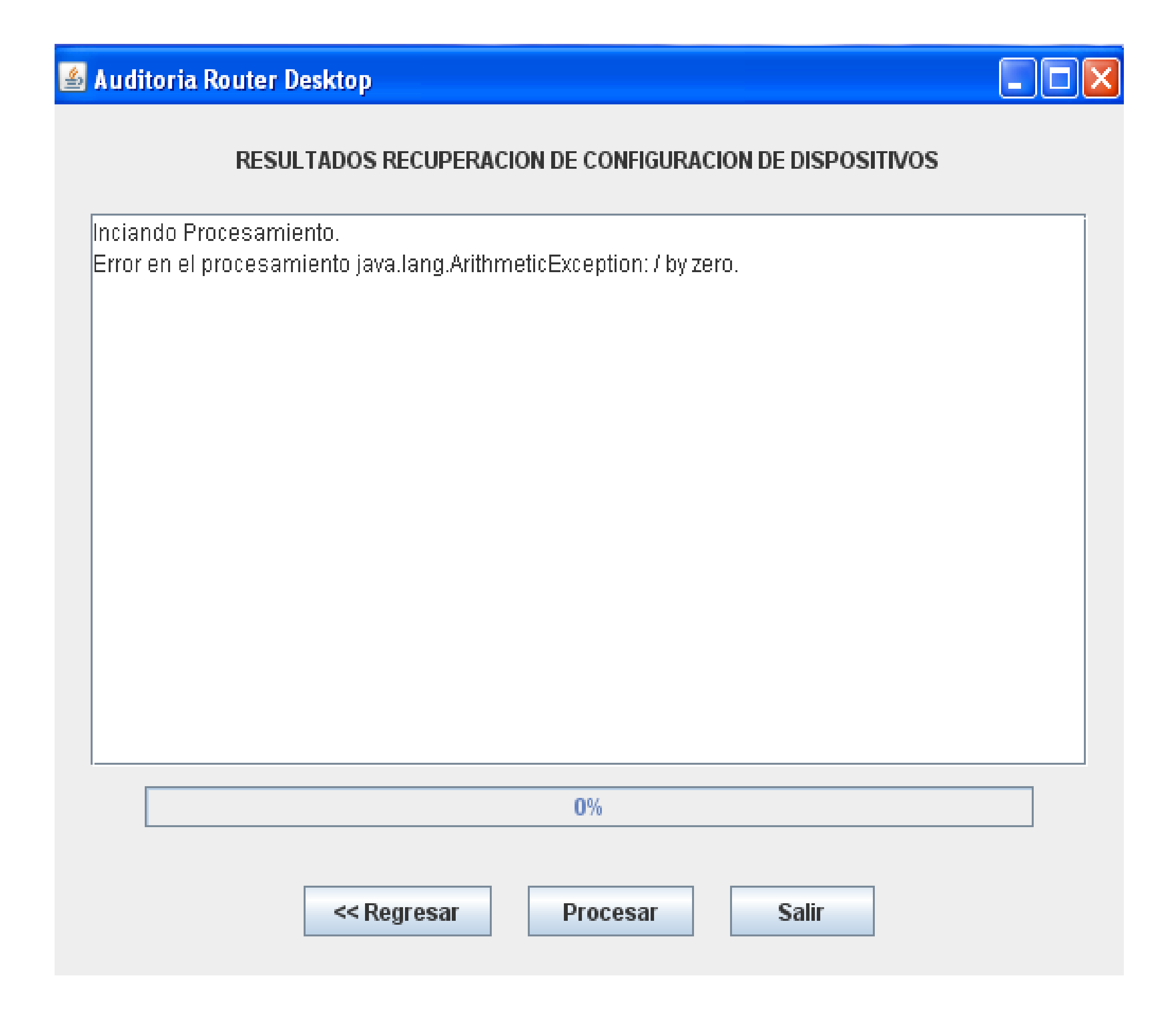


Figura E Extracción de Datos

La extracción de datos vía telnet o TFTP la realiza el auditor para poder realizar vía telnet tenemos que estar conectados al router o switch de la empresa a ser auditada, en

la forma TFTP tenemos que estar en la red de ellos para realizarla la extracción ya que es un archivo con extensión .log que se encuentra en una carpeta llamada

2.9.3 Importación de Datos (HTTP)

La importación de datos se la realiza vía Http en forma inmediata en el momento que presionamos procesar.

2.9.4 Auditoria

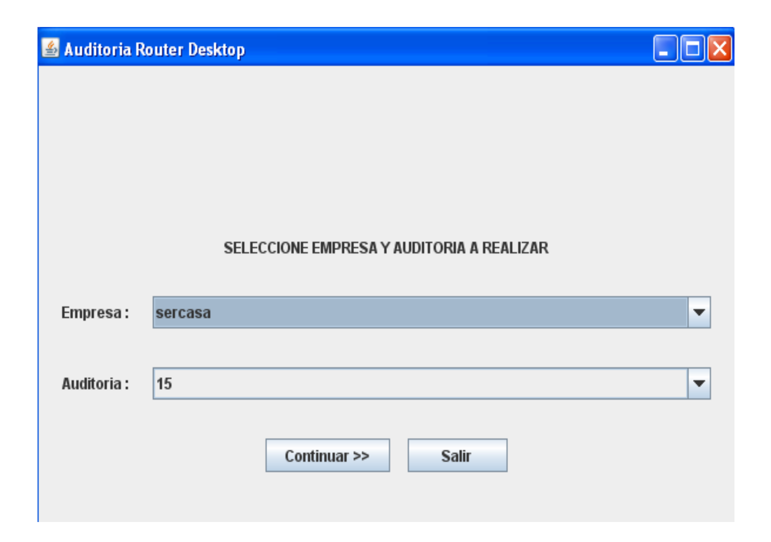


Figura F Auditoria

Aquí el auditor escoge la empresa y la auditoria previamente asignada por el administrador luego presionamos el botón continuar.

2.9.5 **Salir**

Indica la salida del sistema

3.0 **Mantenimiento**

3.0.1 **Empresas**

Crear Empresas



Figura G Crear Empresas

Aquí nos va pedir el nombre, el país provincia, ciudad dirección, el contacto y teléfono de la empresa a ser auditada

Editar Empresa

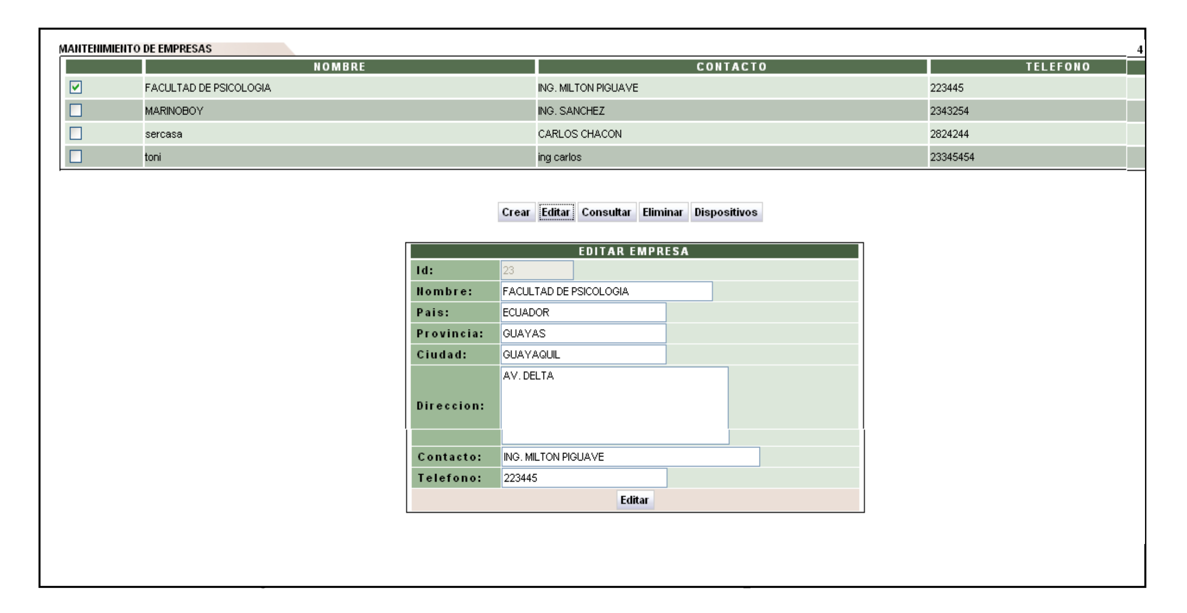


Figura H Editar Empresas

Empresa y podemos modificar datos y guardándolos al dar click al botón inferior llamado editar.

Consultar Empresa

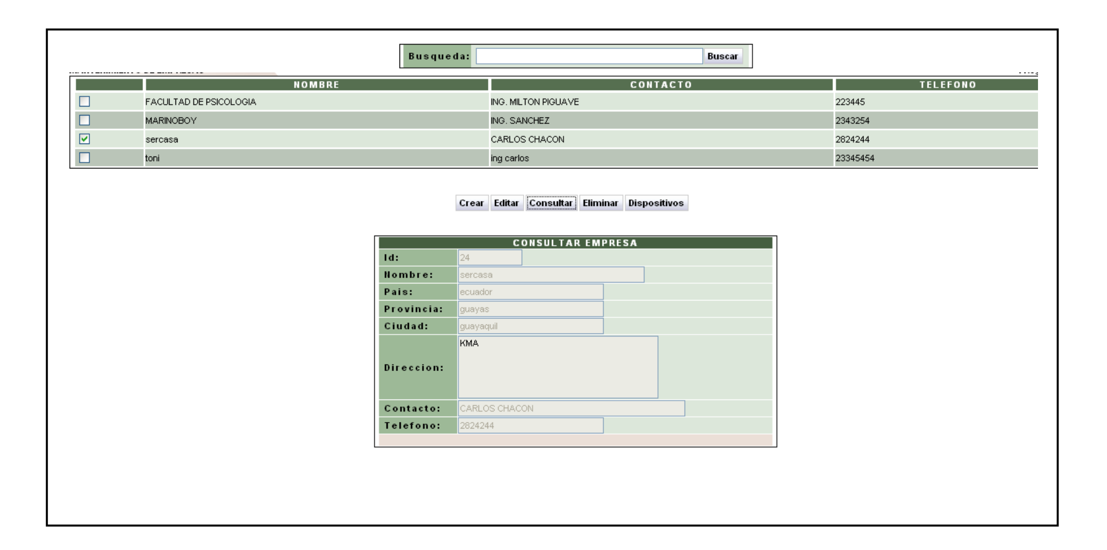


Figura I Consultar Empresas

Primeramente se selecciona la empresa a ser consultada saliendo la información que está en ella.

Eliminar Empresa

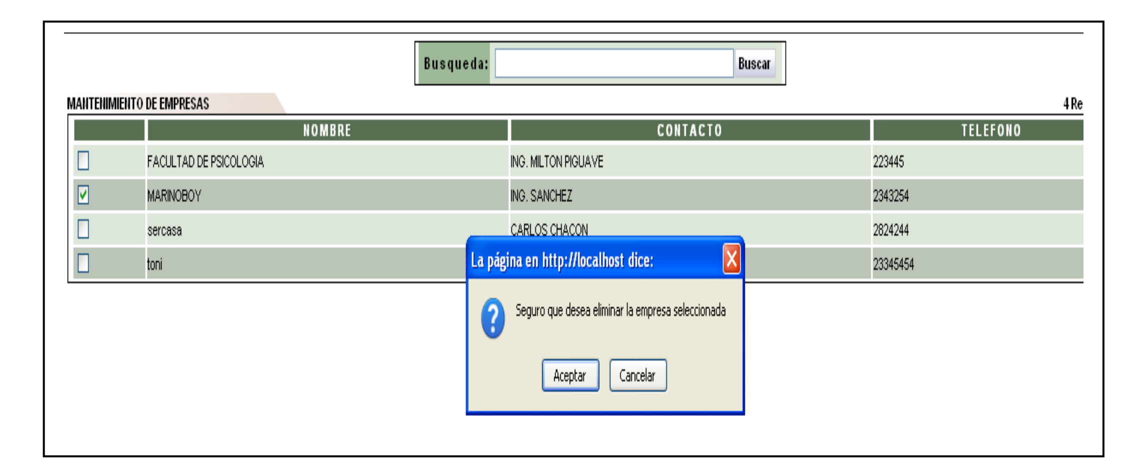


Figura J Eliminar Empresas

Se escoje la empresa a ser eliminada dándole un visto y se da un click en el botón eliminar saliendo un mensaje previo a eliminación.

Dispositivos

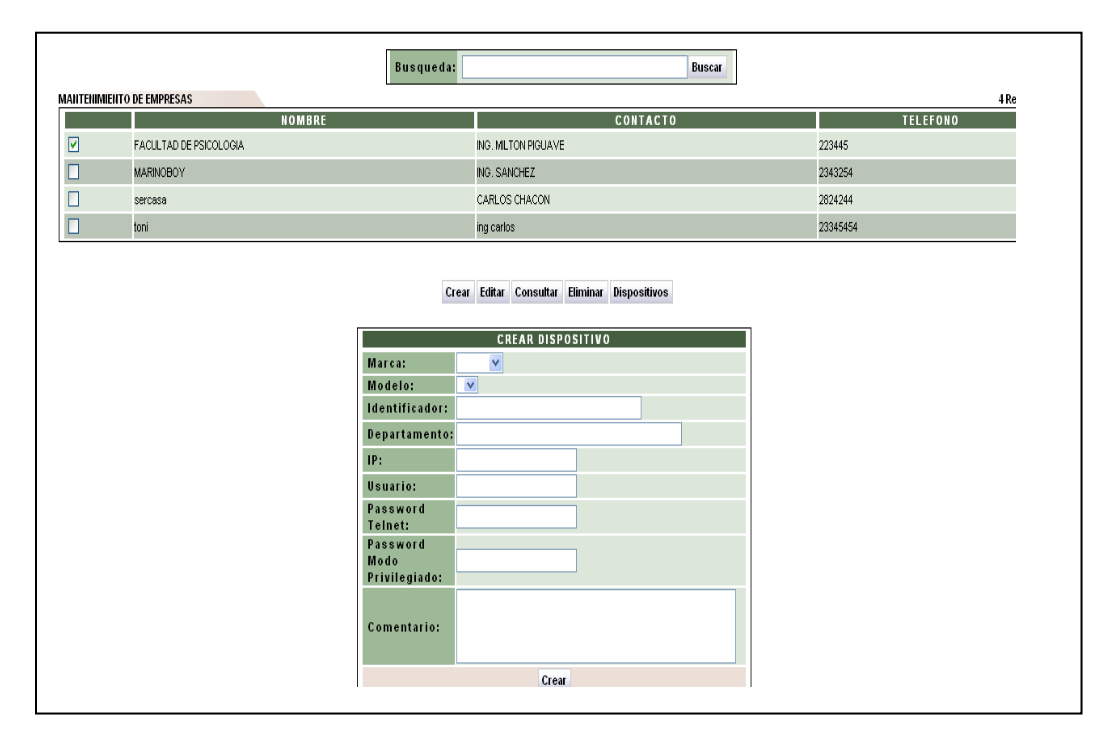


Figura K Dispositivos

Primero seleccionamos la empresa para ubicarles los dispositivos a ser auditados pidiéndonos marca, modelo creados previamente, identificador del dispositivo, la dirección ip que nos debe de facilitar la empresa a ser auditada, usuario del dispositivos si es vía telnet con su respectiva contraseña de este y de modo privilegio si es vía TFTP no se necesita poner estos campos luego aplastamos el botón crear.

3.0.2 **Marcas**

Crear Marcas

Crear Editar Consultar Eliminar

CREAR MARCA

Nombre:

Crear

Figura L Crear Marcas

Para crear marcas debemos de aplastar en el menú la opción marcas y luego nos saldrá una botón que dice crear le damos click pidiéndonos el nombre de la nueva marca damos click en crear

Consultar Marcas

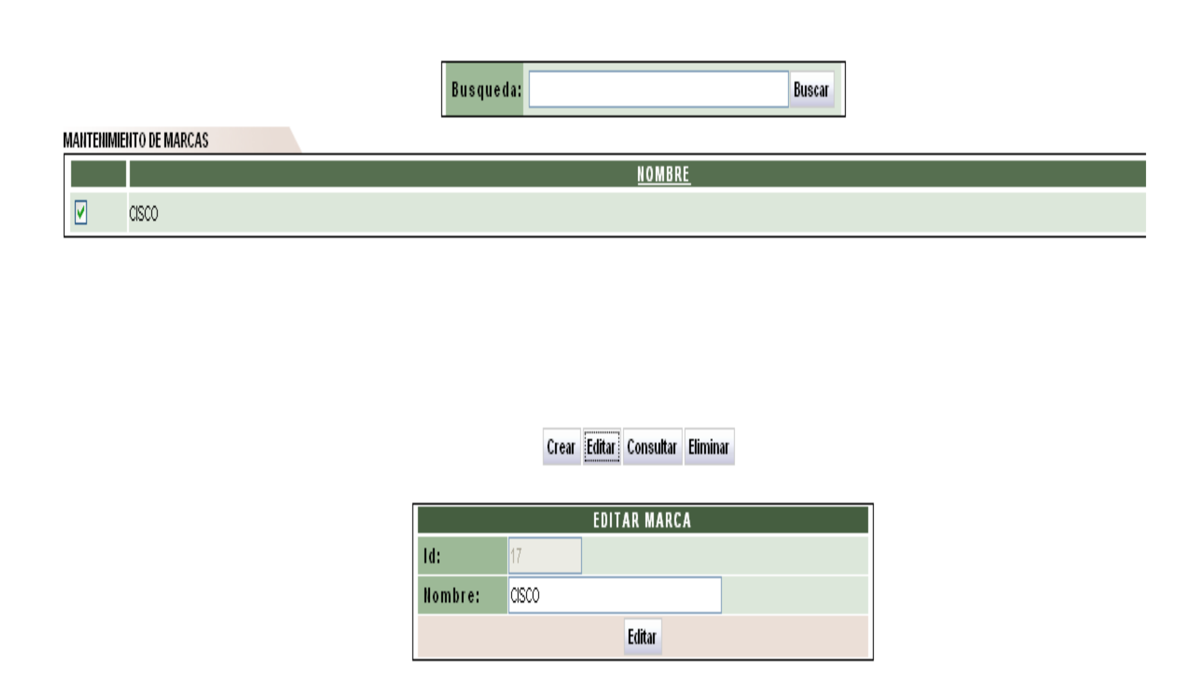


Figura M Consultar Marcas

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos las marcas creadas seleccionando cuál de ellas voy a consultar y le doy click en el botón consultar

Editar Marcas

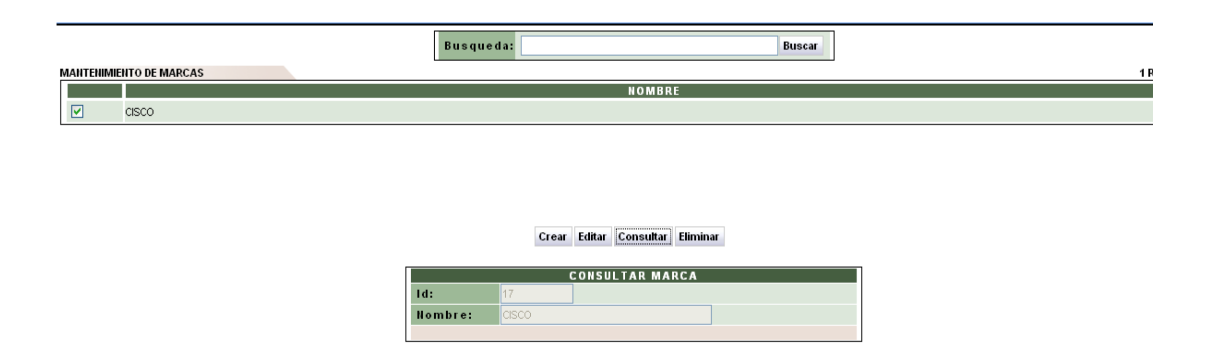


Figura N Editar Marcas

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos las marcas creadas seleccionando cuál de ellas voy a editar y le doy click en el botón editar aquí puedo hacer cambios luego doy click en el botón editar inferior y se guardan los cambios

Eliminar Marca

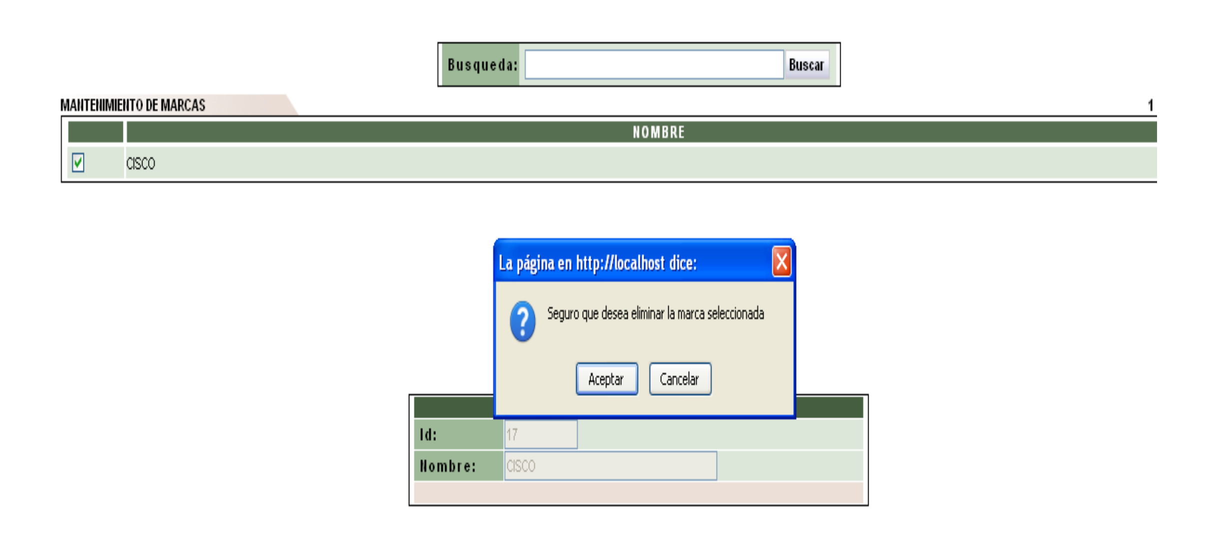


Figura O Elminar Marcas

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos las marcas creadas seleccionando cuál de ellas voy a eliminar y le doy click en el botón eliminar apareciéndome un mensaje previa eliminación que me indica si estoy seguro de eliminar.

3.0.3 Modelos

Crear Modelos

Crear

CREAR MODELO

Nombre:	
Marca:	
Tipo Dispositivo:	

Crear

Figura P Crear Modelos

Para crear modelos debemos de aplastar en el menú la opción modelos y luego nos saldrá una botón que dice crear le damos click pidiéndonos el nombre, la marca previamente creada y el tipo sea router o switch del nuevo modelo damos click en crear.

Editar Modelos

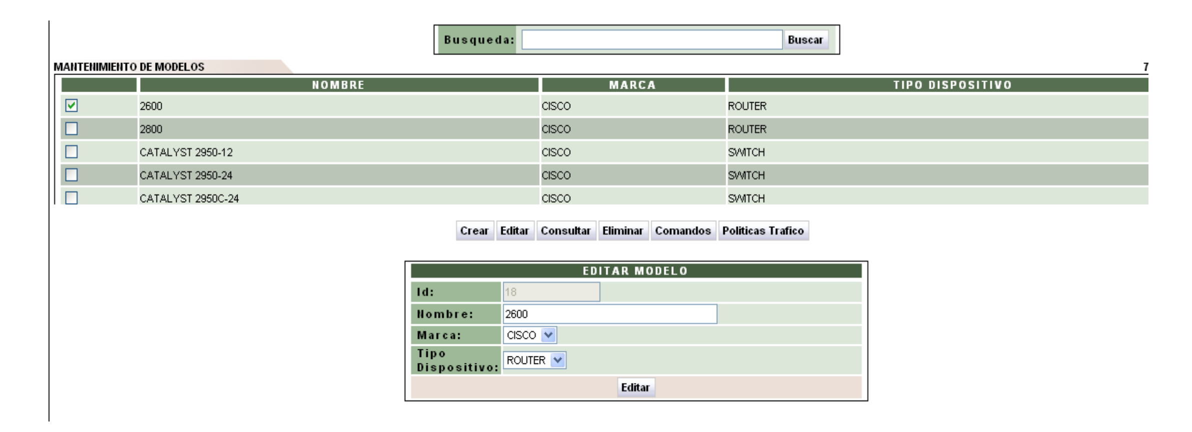


Figura Q Editar Modelos

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos los modelos creados seleccionando cuál de ellos voy a editar y le doy click en el botón editar aquí puedo hacer cambios luego doy click en el botón editar inferior y se guardan los cambios

Consultar Modelos

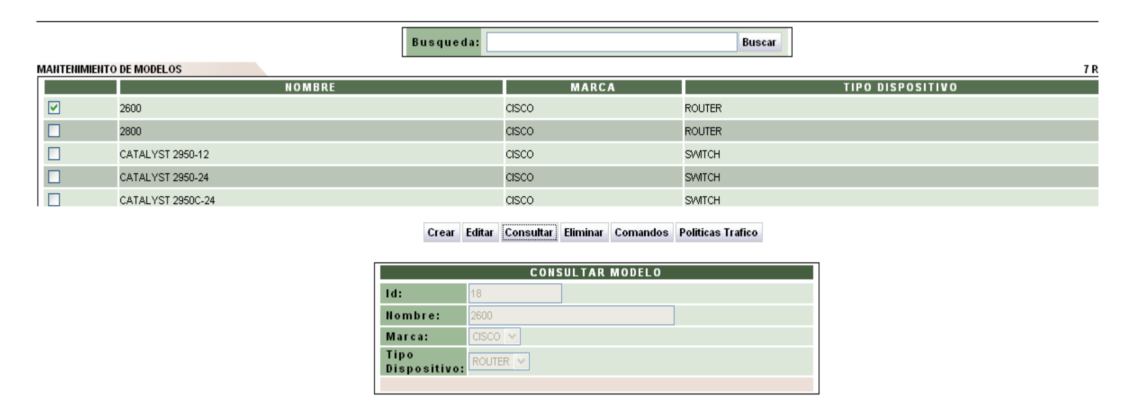


Figura R Consultar Modelos

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos los modelos creados seleccionando cuál de ellas voy a consultar y le doy click en el botón consultar

Eliminar Modelos

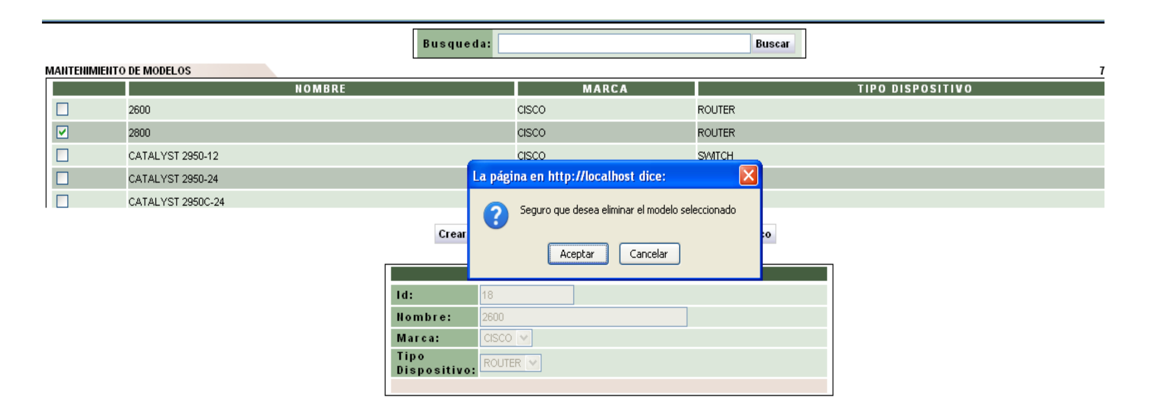


Figura S Eliminar Modelos

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos los modelos creados previamente seleccionando cuál de ellos voy a eliminar y le doy click en el botón eliminar apareciéndome un mensaje previa eliminación que me indica si estoy seguro de eliminar.

3.0.4 Comando y Políticas de Dispositivo

Comandos de dispositivo

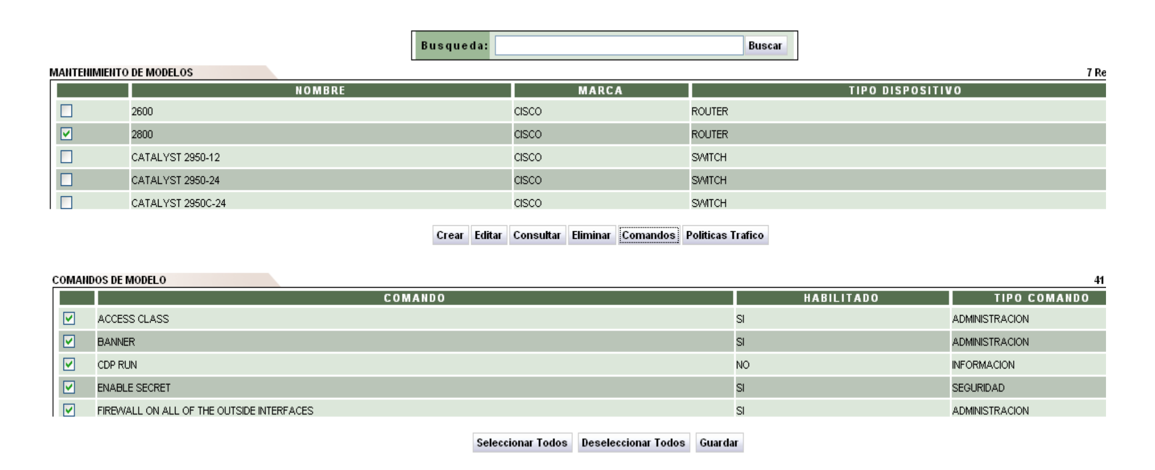


Figura T Comandos de Dispositivo

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos los modelos creados previamente seleccionando cuál de ellos voy a asignarle los comandos y le doy click en el botón comandos apareciéndonos comandos previamente creados en los cuales puedo seleccionar todos o algunos y también deseleccionar todos luego damos click en el botón guardar y en ese momento ya están asignados comando para ese modelo

Políticas del dispositivo

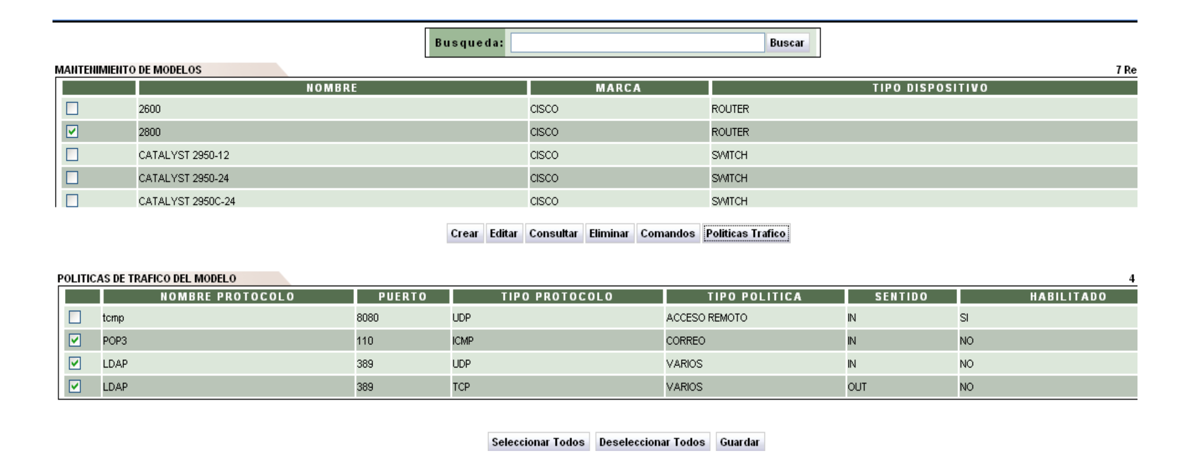


Figura U Políticas del dispositivo

Primeramente debemos de dar click en buscar o poner el nombre en la parte superior donde está el botón búsqueda apareciéndonos todos los modelos creados previamente seleccionando cuál de ellos voy a asignarle las políticas de tráfico y le doy click en el botón comandos apareciéndonos políticas previamente creadas en los cuales puedo seleccionar todos o algunos y también deseleccionar todos luego damos click en el botón guardar y en ese momento ya están asignadas las políticas de tráfico para ese modelo

3.0.5 Audioría

Crear Auditoria

Crear

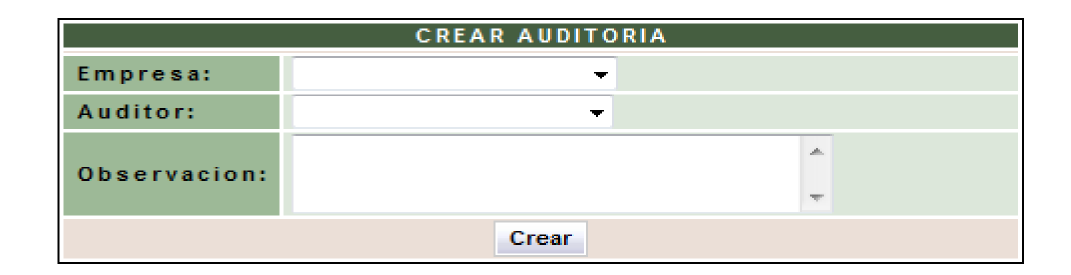


Figura V Crear Auditoria

Para crear una auditoria se debe dar click en el menú en donde dice auditoria se da click en el botón crear y nos pide el nombre de la empresa a ser auditada, la persona que se va asignara para que realice la auditoria y una observación le damos click en el botón inferior crear y se guarda

Consultar Auditoria

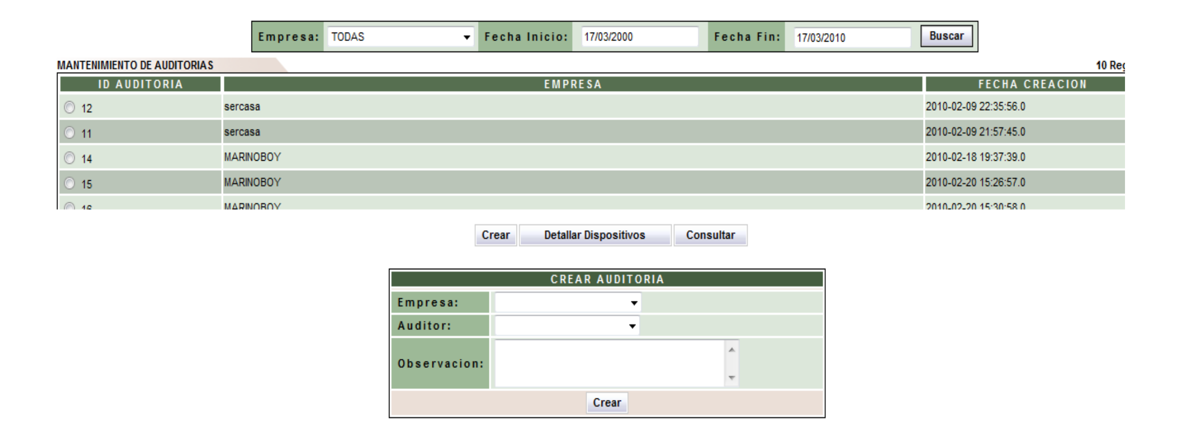


Figura W Consultar Auditoria

Para consultar auditoria asignadas se debe ´poner el nombre de la empresa o la fecha desde/hasta luego se marca la auditoria a ser consultada y se da click en crear

Detalle Dispositivo

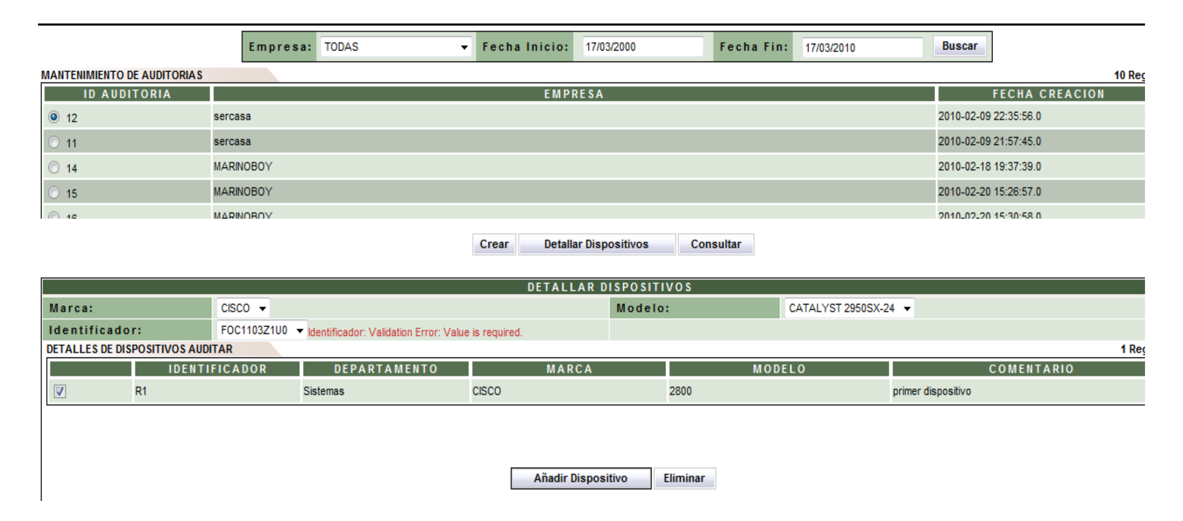


Figura X Detalle Dispositivo

Para detallar los dispositivos que se van a ser auditados primeramente se debe se señalar la empresa, se da click en el botón Detalle Dispositivo y nos aparecerá pidiéndonos la marca, modelo e identificación del dispositivo, nos aparecerán los dispositivos de esa empresa, debemos de marcar cuales dispositivos deseamos que se auditen y damos click en el botón añadir dispositivos.

3.0.6 Comandos

Crear Comando

Crear



Figura Y Crear Comandos

Para crear comandos debemos de dar click en el menú opción comandos luego damos click en el botón crear nos pide tipo de comando, el nombre del comando, dos mensaje que realiza ese comando y el estado habilitado sí o no damos click en el botón inferior crear

Editar Comandos

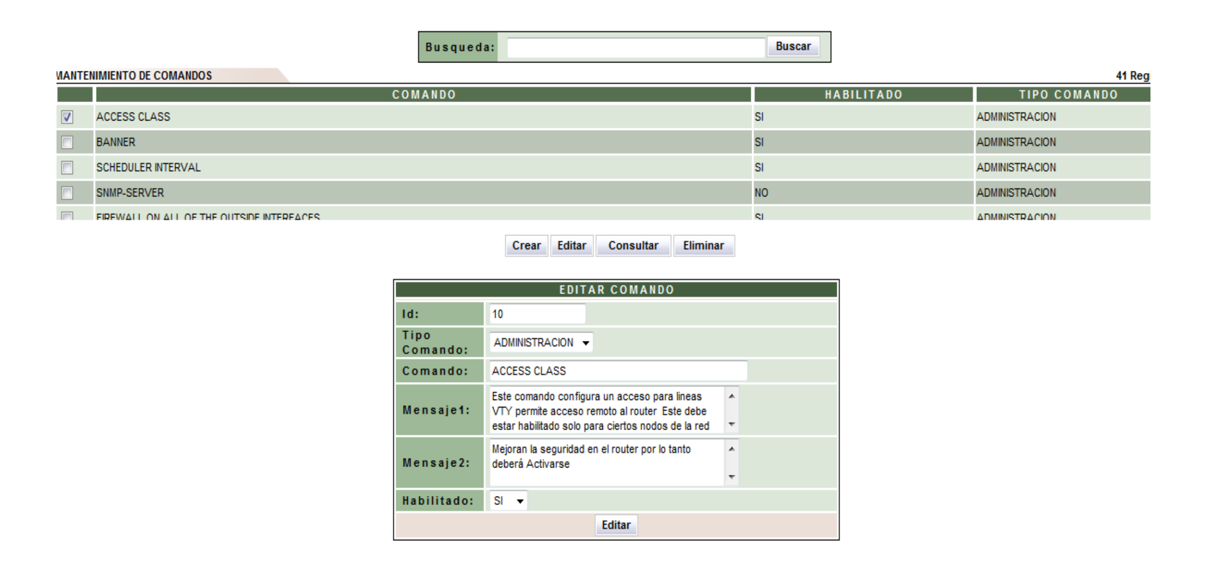


Figura Z Editar Comandos

Primero debemos de buscar el comando a ser editado poniendo su nombre y aplastando el botón buscar o tan silo el botón buscar y nos aparecerá los comandos creados seleccionamos el comando a editar y damos click en el botón editar aquí podemos hacer cambios y luego para grabarlos damos click en el botón editar que está en parte inferior

Eliminar Comandos

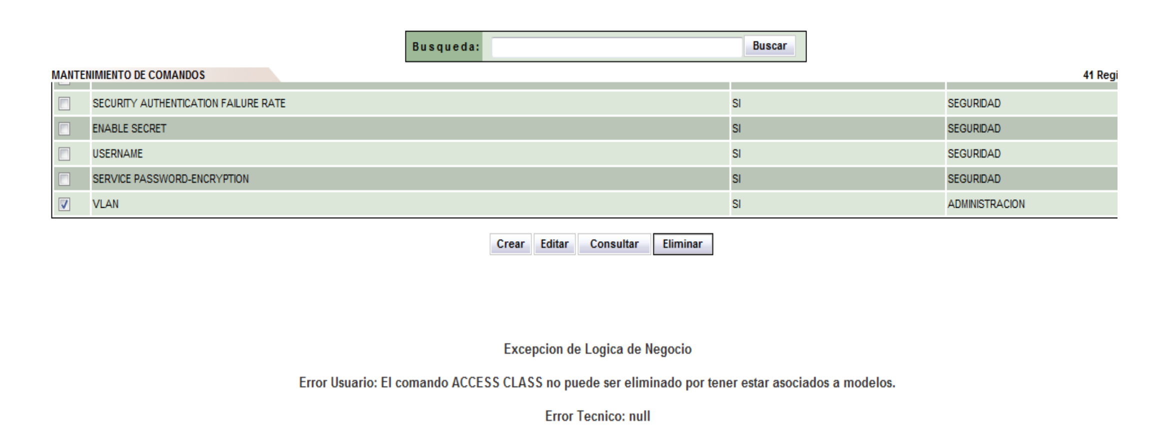


Figura AA Eliminar Comandos

Para eliminar comandos debemos primeramente señalarlos luego damos click en eliminar si acaso estos comandos estan ligados a audiorias entonces no se podrán eliminar.

3.0.7 Políticas Trafico

Crear Políticas

Crear

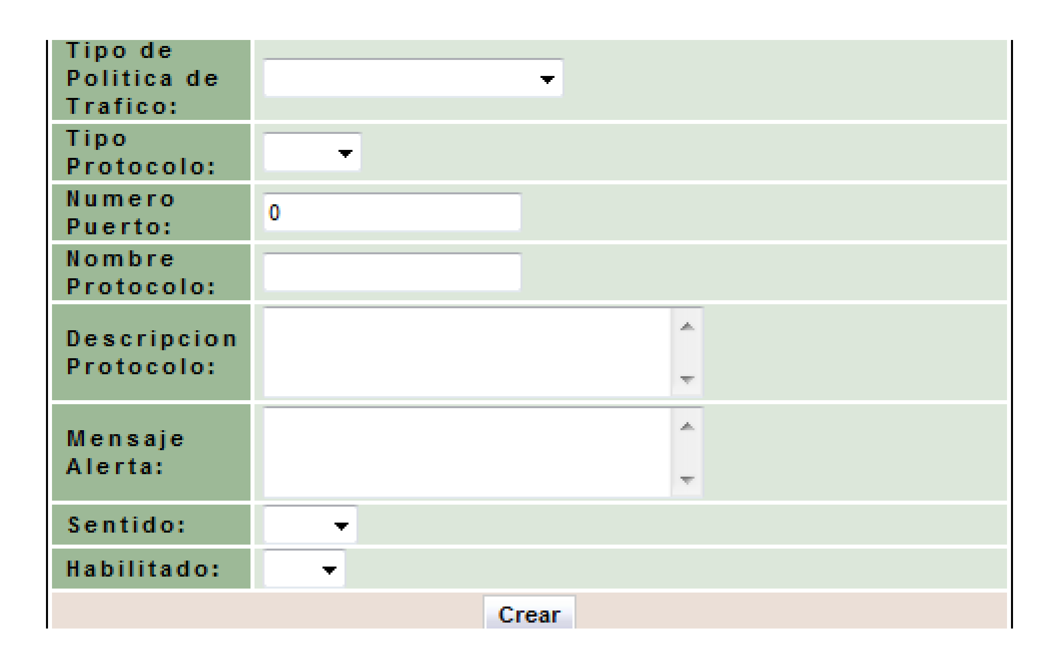


Figura AB Crear Políticas

Para crear políticas de tráfico debemos de dar click el menú políticas de tráfico y dar después click en el botón crear y nos pide el tipo de policía, tipo de protocolo, número de puerto, nombre del protocolo, descripción del protocolo, mensaje de alerta, sentido si es de entrada (in) y de salida (out) y si está habilitado si o no

Editar Políticas



	TIPO PROTOCOLO	NOMBRE PROTOCOLO	NUMERO PUERTO	SENTIDO	HABILITADO	TIPO POLITICA
☑	ICMP	POP3	110	IN	NO	CORREO

Crear | Editar | Consultar | Eliminar

Tipo de Politica de Trafico:	CORREO
Tipo Protocolo:	ICMP
Numero Puerto:	110
Nombre Protocolo:	POP3
Descripcion Protocolo:	Protocolo para mensajeria
Mensaje Alerta:	Principalmente utilizarlo para correo interno
Sentido:	IN
Habilitado:	NO

Editar

Figura AC Editar Políticas

Para editar políticas primeramente debemos de buscarla podemos poner el nombre del protocolo (servicio) o el numero de puerto damos click en el botón buscar y nos parecerá las políticas y señalamos la que deseamos editar y le damos click en el botón editar aquí podemos cambiar el contenido y para grabar le damos click en botón editar que se encuentra en la parte inferior.

Consultar Políticas de Trafico

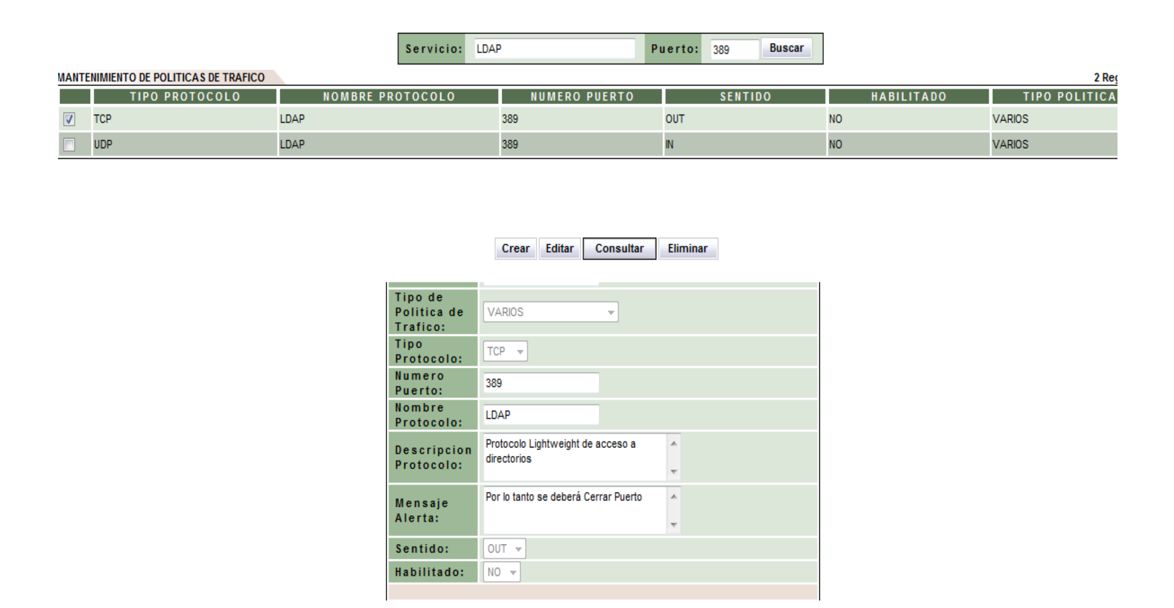


Figura AD Consultar Políticas de Trafico

Para consultar políticas debemos de primeramente señala la política a ser consultada luego le damos click en el botón consultar y nos aparecerán los datos de esa política de tráfico

Eliminar Políticas Trafico

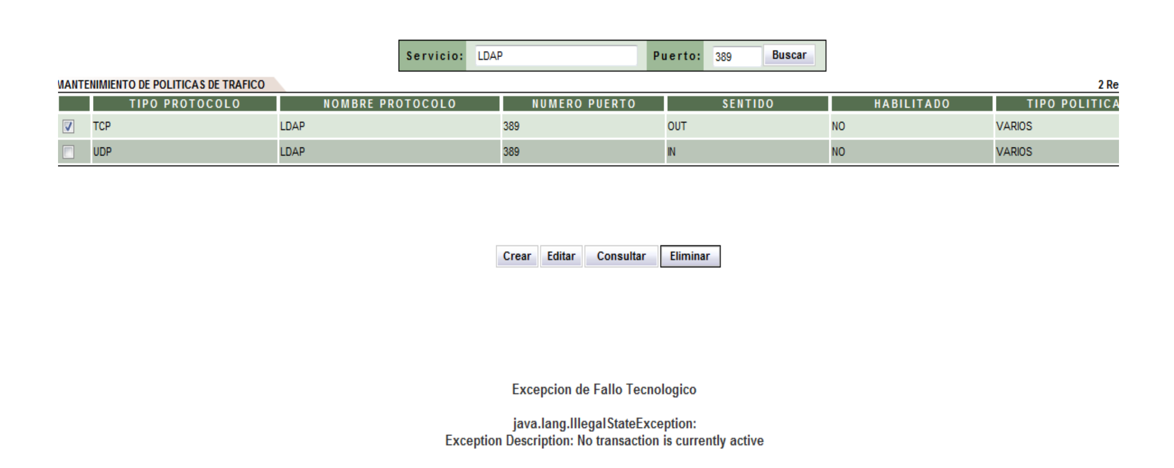


Figura AE Eliminar Políticas de Trafico

Para eliminar políticas de tráfico debemos de señalarla y si dar click en el botón eliminar y si acaso esta está ligada algún dispositivo entonces no se podrá eliminar

3.0.8 Usuarios

Crear Usuarios



Figura AF Crear Usuarios

Para crear usuarios debemos de dar click en el menú en donde dice usuarios y nos aparecera un botón crear le damos click y nos pide ingresar el usuario, clave, nombres , apellido paterno, apellido materno y su estado activi si o no.

Editar Usuarios

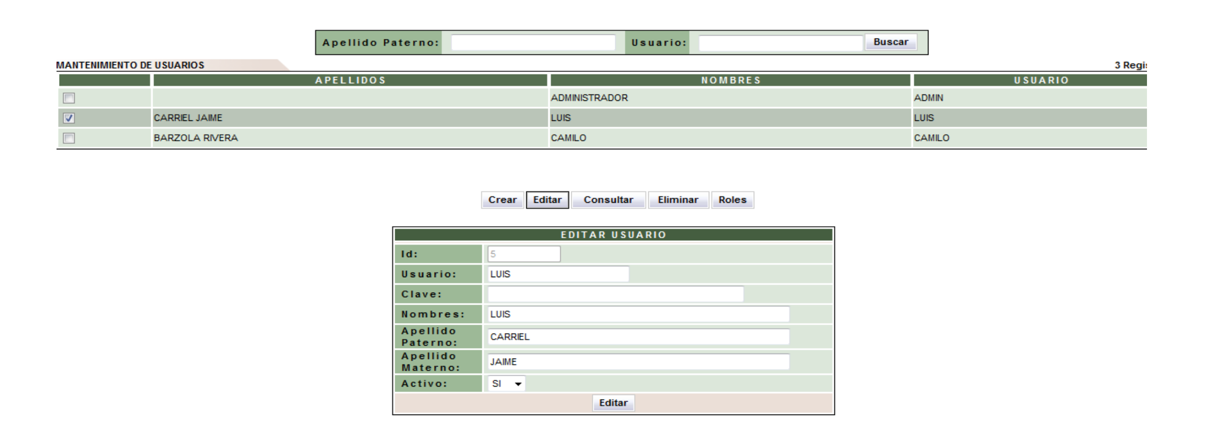


Figura AG Editar Usuarios

Debemos de buscar el usuario a ser editado poniendo su apellido paterno o el usuario dándole click en el botón buscar nos aparecen los usuarios señalamos el usuario a ser editado y damos click

Consulta Usuarios

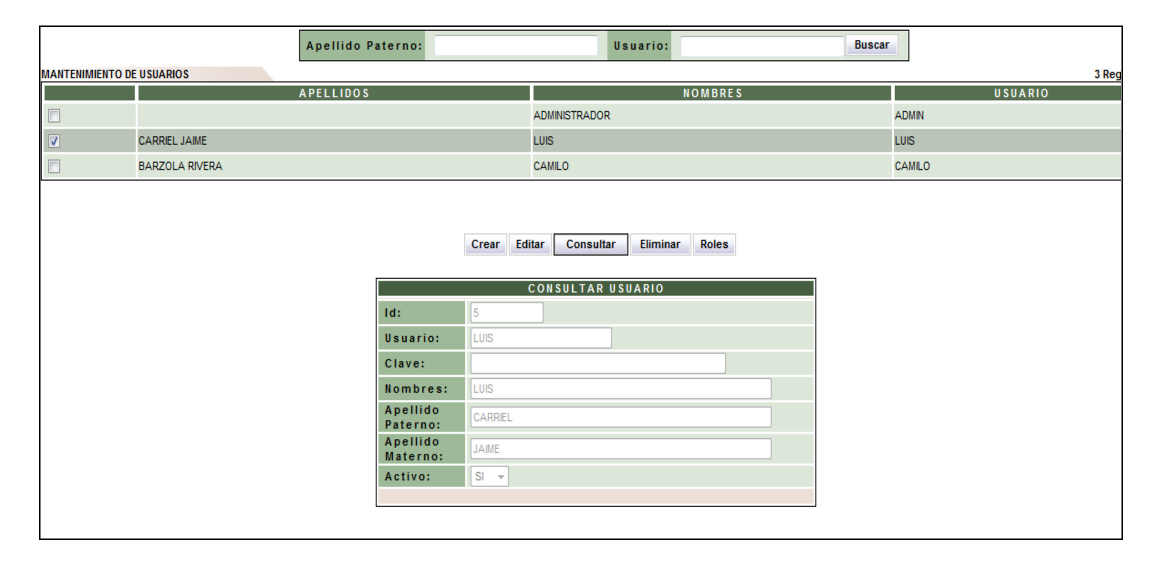


Figura AH Consulta Usuarios

Para consultar primero señalamos cual va a ser el usuario que deseamos que aparezcan los datos luego lo marcamos y damos click en el botón consultar.

Elimina Usuarios

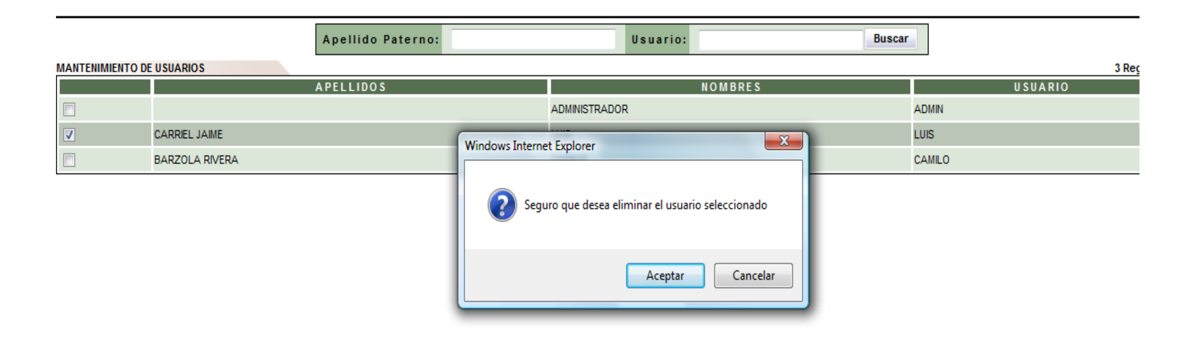


Figura AI Eliminar Usuarios

Para eliminar un usuario primeramente se lo señala y se da click en eliminar si ya hay auditoria declaradas par él no se va a poder eliminar.

Roles a los Usuarios

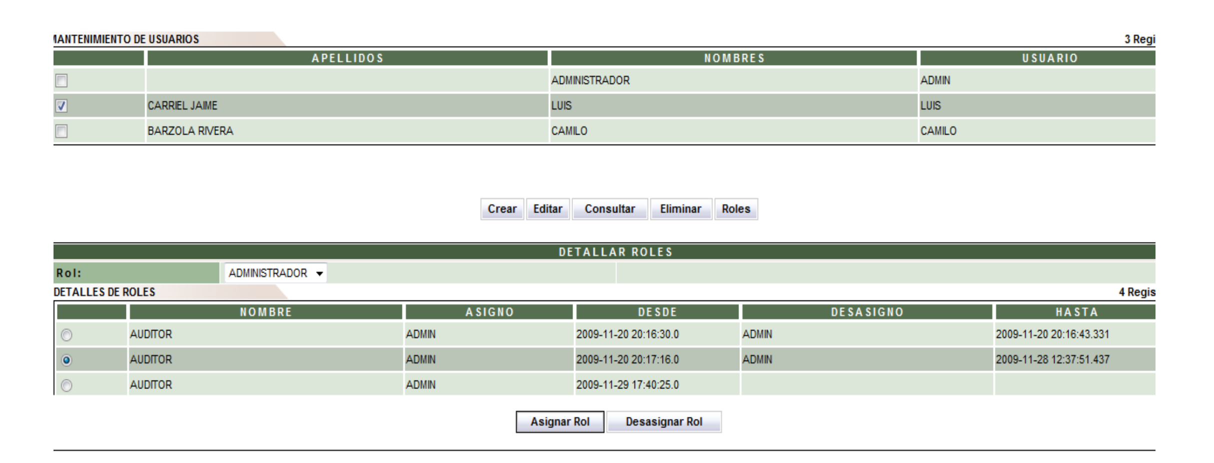


Figura AJ Roles a los Usuarios

Nos permite asignarle el rol que va a cumplir el usuario primeramente se escoge al usuario y se da click en el botón roles nos indica en forma detallada los usuarios creados y escogemos el rol asignarle y amos click en el botón asignar rol.

Salir de Sesión

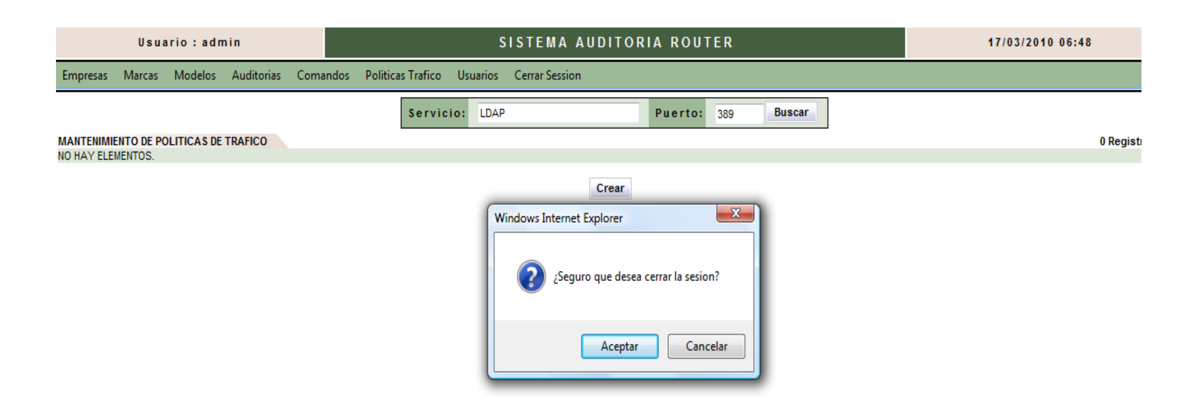
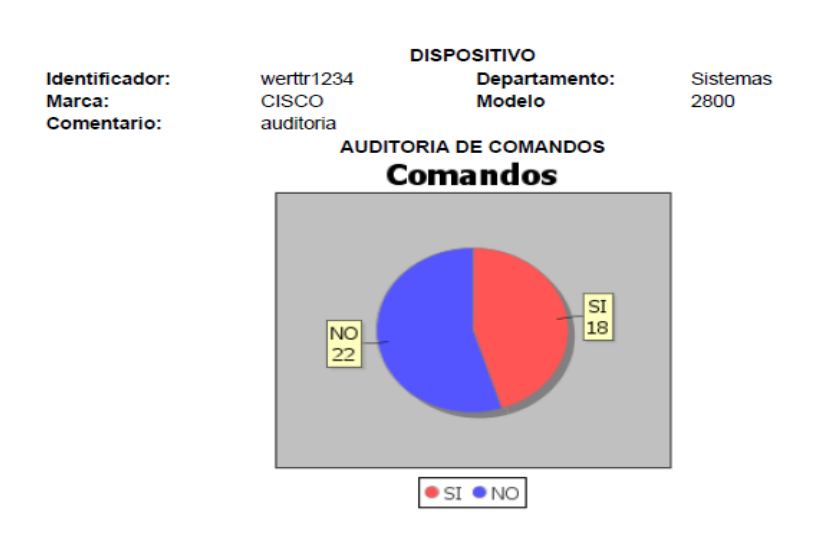


Figura AK Salir de Sesión

3.1 Reporte

3.1.1 Reporte de Auditorías realizadas



Comando	Habilitado	Cumplio	Definicion	Recomendacion
IP REDIRECTS	NO	SI	Este comando trasmite informacion acerca de la vias rutas y condicciones de la red.	En una red IP que funcione correctamente, un router enviará redireccionamientos solamente a los hosts de sus propias subredes locales ningún nodo final enviará un redireccionamiento y ningún redireccionamiento se atravesará más de un

Figura AL Reporte de Auditorías realizadas

Aquí se muestra el resultado de la auditoría mostrando la imagen con el número de comandos o políticas que cumplió dentro de la auditoria, así como

también la definición del comando o política de tráfico con su respectiva recomendación.

BIBLIOGRAFÍA

http://www.cisco.com: Página Oficial de Cisco

http://www.microsoft.com/windowsserver2003/default.msp : Página Oficial de Windows Server 2003

http://tomcat.apache.org : Pagina Oficial de Tomcat

http://commons.apache.org/net : Pagina Oficial para Librerías de telnet

http://itextpdf.com : Pagina para Crear reportes en Java

Guía del usuario de Cisco Router and Security Device Manager

Software de Pruebas: Cisco Packet Tracer